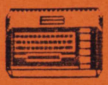
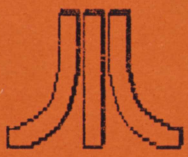
T—Y—N—E—&—W—E—A—R



TYNE & WEAR



ATARI 8-BIT USER GROUP

Newsletter of TWAUG

Software
Editorial
Buy & Sell
Hardware

Reviews
Help line
Section
Repair Info

Public Domain Library

ISSUE #7

JANUARY/FEBRUARY 1994

U—S—E—R—G—R—O—U—P

BRING YOUR EIGHT UP TO DATE
with power products from
COMPUTER SOFTWARE SERVICES

THE BLACK BOX

The BLACK BOX is an add-on board for the Atari 600XL,800XL and 130XE 8-bit computers. It is a T-shaped board that plugs into the PBI port of the XL computer, or the ECI and cartridge ports of the 130XE. Connectors for both types of computers are built into the BLACK BOX so no adapter boards are necessary. A cartridge port is available on the board itself for 130XE users.

The BLACK BOX provides many unique and useful functions. The four primary functions are:-

* RS-232 serial modem port
* Parallel printer port
* SASI/SCSI hard disk port
* Operating System enhancements

The BLACK BOX is $199.95 for the basic unit, and $249.95 with an onboard 64k printer buffer.
Shipping and Handling extra.

THE BLACK BOX ENHANCER

A must for all BLACK BOX owners. The BLACK BOX ENHANCER is a plug-in module for your BLACK BOX, enhancing the printer functions and adding an instantly available, full featured sector editor!

Installation of the BLACK BOX ENHANCER requires one simple solder connection. Only $49.95 plus shipping/handling.

THE FLOPPY BOARD

Our latest and greatest product. The FLOPPY BOARD is an add-on expansion board for the BLACK BOX interface. It allows the use of the same inexpensive floppy drive mechanisms used in IBM computers. The FLOPPY BOARD is the first floppy drive interface to support "high density" floppy drive mechanisms in either 5.25 inch or 3.5 inch. Built into the FLOPPY BOARD are our BLACK BOX ENHANCER and a version of our SUPER ARCHIVER to allow copying of protected disks for 3.5 inch format. Included with the FLOPPY BOARD is our program to read and write to IBM or ST formatted disks. This makes the FLOPPY BOARD the best way to transfer files to and from your 8-bit.

The FLOPPY BOARD is only $149.95 plus shipping & handling.

THE MULTIPLEXER

This device brings the power and flexibility of larger systems to your 8-bit. The Multiplexer is a collection of cartridge interface boards that allow up to 8 Ataris to read and write to the same drives (typically a hard disk), access the same printer(s), and talk to each other. It is the first practical networking system for the Atari 8-bit computer.

One "master" computer (any 8-bit) is equipped with the master Multiplexer interface. Then up to 8 "slave" computers hook up to this master, each having their own slave interface.

The "common" peripherals (things that are to be shared) are connected to the master. On each slave, all disk and printer I/O is routed through the master, so no extra disk drives are needed.

The Multiplexer sells for $199.95 for a master and two slave units with cable. Additional slave units are $89.95 each, plus shipping/handling.

THE SUPER ARCHIVER II

The SUPER ARCHIVER II edits and copies all enhanced density programs plus retains all the features of the SUPER ARCHIVER.

The SUPER ARCHIVER II is only $99.95 plus shipping & handling. NOTICE: if you already have THE SUPER ARCHIVER you may upgrade to S.A.II for only $29.95 plus shipping/handling. Software only.

THE BIT WRITER

The Super Archiver BIT WRITER is capable of duplicating even the "uncopyable" Electronic Arts and Synapse Syn-series, which employ 34 full sector tracks. The BIT WRITER must be used with the SUPER ARCHIVER

The BIT WRITER is only $79.95 plus shipping/handling.

THE ULTRA SPEED PLUS OS

The Operating System that should be in every XL/XE computer! The Ultra Speed Plus puts unbelievable speed and convenience at your fingertips.

Use any DOS to place Ultra Speed formats on your disks (with XF551 or modified 1050 drives), reading and writing at this speed with most programs. This high speed mode can be turned off for maximum compatibility.

Four simple solder connections are required for installation if your machine has a socketed OS ROM. The Ultra Speed OS is only $69.95 plus shipping/handling.

For more information on these and other 8-bit products:

CONTACT
COMPUTER SOFTWARE SERVICES
PO BOX 17660
ROCHESTER, NEW YORK 14617
USA

ORDERING LINE: (716) 429-5639
FAX: (716) 247-7158
BBS: (716) 247-7157

or contact T.W.A.U.G. we will do our best to help.

EDITORIAL

Who to blame!!!

John Matthewson
David Ewens
Max Gerum

NEW YEARS GREETINGS!

John, David and Max would like to wish all TWAUG readers the very best of everything, good times, good luck and happy computing for 1994.

A BIG THANK YOU from Max

I always knew that all Atarians were very friendly, shared their 8-bit knowledge with everyone else and made you feel as if it was a big family. But what makes me feel proud and really touch a soft spot in my heart is the fact that everyone cares for others. I received a lot of Get Well messages after my operation, David passed them on to me and I also received a lot personally and therefore my Thank You to everyone is specially big. Sorry but I am unable to thank you all personally, if I did, I don't think I would be able to write this newsletter.

I am now back again behind my desk tap, tap, tapping away at the keyboard and by doing this I am sure I will soon feel one hundred percent again.

There is only one small apology I must make, before I went into hospital I left my colleagues to finish off the newsletter and of course they had a lot of extra work to do. By doing that, David wrote in the apology paragraph the hospital was in NEWDASTLE instead of NEWCASTLE. I thought I mention this and rectify it just in case someone thinks that there is a place called Newdastle.

The next issue will be ready by mid-March.

CONTENTS

EDITORIAL 3

THE FLOPPY BOARD
accessory for Black Box
Reviewed by C.Koontz 4

PUTER PROGRAM
A typing program for the younger member of the family
by Ed.Hall 6

HINTS & TIPS
by D.Ewens & B.Jackson 7

MARK'S GAMES COLUMN
by Mark Stinson 8

ANYBODY LISTENING & PORT85
Secret of talking to your computer
by Jim Cutler 10

THE PARALLEL BUS
by Earl Rice 11

LETTER SECTION
and BUY & SALE 14

XF551 UPGRADE
by J.Dickerson 15

REVIEWS
by Mark Fenwick 16

FIVE LINERS 18

CRACKING THE CODE
Machine code for beginner 19

DISK CONTENT 23

DAISY DOT 3 & PAGE 6 WRITER
by John Bunting 24

TEXTPRO PERFECT WORDPROCESSOR
by Max Gerum 25

TWAUG MEMBERS PHOTOS FROM SHOW 28

PICTURE MANIPULATOR
by Ed Hall 29

THE FLOPPY BOARD

Connecting any Disk Drive to Your 8-Bit

By Charlie Koontz

This article first appeared in "Current Notes" magazine Your Monitor on the world of Atari Dec92/Jan 93 issue

Floppy board? Is that some kind of rubber circuit I can press into any space in my computer? Nah. The Floppy Board, from always innovative Computer Software Service, is just the best accessory introduced this year for the Atari 8-bit computers. When added to your Black Box, also from Computer Software Services, the Floppy Board gives you the ability to run generic floppy drives on your XL/XE computer, at parallel bus speed. Also included as part of the Floppy Boards firmware is the "Taskmaster" sector editor and a version of the "Super Archiver".

What does this mean? The Floppy Board (let's just call it "FB") allows you to run those high density floppy drives previously available only to 16 and 32 bit users. It's basically a controller board designed to let you attach virtually any bare floppy drive an IBM PC or ATARI ST can use. Which drives? just about any 3 1/2-inch or 5 1/4-inch drive you can name. Even those old 8-inch drives can be used. See Table 1 for specifics.

Table 1. Floppy Drive Types

Size	Density	Sides	Capacity	Sectors	Tracks/side
5.25"	single	1	90K	720	40
5.25"	double	1	180K	720	40
5.25"	double	2	360K	1440	40
5.25"	quad	2	720K	2880	80
5.25"	high	2	1200K	4640	80
3.5"	double +	1	360K	1440	80
3.5"	double	2	720K	2880	80
3.5"	high	2	1440K	5760	80
8.0"	high*	2	1000K	4004	77

* - featured on early Atari St computers.

Since the FB is a parallel bus device, the drives read and write at blinding speed. While they wont be quite as fast as a hard disk drive, it's the next best thing. Up to four drives can be run from the FB. This is in addition to any floppy drives attached to your computer's serial I/O port, so you could be running as many as eight floppy drives from your Atari, four from the serial port and four from the parallel port via the FB.

INSTALLATION

The FB attaches directly to the Black Box. In fact the FB circuit board is installed upside down on top of the Black Box circuit board, covering about two-thirds of the right-hand side of the Black Box. To do this, you must first remove the original Black Box 28-pin ROM chip and replace it with the new ROM accompanying the FB; the new ROM has two wires extending from it that must be soldered into place on the Black Box circuit board. Next remove the 40-pin chip just to the left of the Black Box ROM you just replaced.

The FB plugs, component down into the Black Box in two places: that empty 40-pin chip socket and the 20-pin single row connector above the toggle switches. A pair of snap supports secure the right side of the FB to the Black Box. The optional plastic case still fits over the FB-Black Box combination; even if, like me, you find it necessary to use a couple of long cable ties to make sure the electrical connections between the two boards don't pull apart slightly. Installation is actually easier than it sounds but some soldering is required.

INITIAL SETUP

Here's the easy part. After installing the FB, set the Black Box DIP switch #2 to OFF. This contradicts the instructions in the Black Box manual but is required by the FB firmware. Evidently, the assumption is that the switch function is not necessary since you will always have a parallel bus drive, either hard or floppy, running from the FB-Black Box combination.

Now you need to connect those generic floppy drives. The Floppy Board output is a 34-connection card edge, the same as the inputs to generic 5.1/4-inch drives so stock up on 34-connection card edge connectors. The small 3.1/2-inch drives requires a 34-pin IDC female connector with two rows of 17 pins each. Press the connectors onto a 34-conductor ribbon cable at the appropriate intervals and you're all set. Be sure to line up pin 1 as instructed in the Floppy Board manual. Be sure to check that each of those device ID shunts on your generic drives is set to a different value between 1 and 4 (or 0-3 for some drives).

Finally, make sure all those generic drives are properly fed. Each drive requires a +5v dc through a 4-pin white plastic-shroud connector. Unfortunately, 5.1/4 inch drives and 3.1/2 inch drives use slightly different connectors; the 3.1/2 inch drive connector is smaller. There are lots of vendors selling drive enclosures with power supplies and power plug adapters to fit your needs.

FOR PERCOMS ONLY

For those of you who are bypassing a Percom controller card to use the bare drive mechanism included in Percom's line of Atari drives, you'll find that the "system" or master drive doesn't let you set a device ID; it's assumed to always be D1:. Sorry, Percom didn't provide any alternatives. You'll also find you can't completely dispense with the Percom controller card; part of the power supply circuitry that provides the regulated +5vdc to the drive mechanism is located on the controller board. Of course, one can always install a different power supply but that's too ambitious for this article. All you can do is run the ribbon cable from the FB to the 34-conducter card edge of the Percom drive mechanism, which is hidden inside the drive case. The card edge you see exposed on the rear of the Percom case is for connecting add-on or "slave" drives to the Percom controller card; Ignore it.

FLOPPY BOARD continued

CONFIGURATION MENU

Once all those drives are hooked up you're ready to roll right? Well, not quite. You've got to configure the FB-Black Box so it knows what you've connected to it. When you enter the Black Box Drive Configuration Menu, You'll see a new option F at the bottom of the screen for Floppy Board Configuration. Once you're in this menu, you can step through the drive types presented in Table 1 for each of the (up to four) floppy drives on the parallel bus. You must also set the step rate and double-side mode (for double sided drives). It sounds complicated but the Floppy Board manual makes it easy. The final manual should be even better than the preliminary manual shipped with the early Floppy Boards.

What is this double-side mode? Well, it appears that there was never any real standard for the configuration of double-sided drives for the Atari. The FB allows setting each parallel floppy drive to one of the following double-side modes: ATR-8000, Percom, or XF-551. The main reason to choose one over the other is to maintain compatibility with one or more drive types. If you have some Percom or ATR-8000 double-sided disks in your library, you'll want to keep at least one parallel floppy drive set to that mode. Incidentally, when Percom drives are used with the FB by bypassing the Percom controller, they are freed from operating only in the Percom mode; they can be set to any of the three double-side modes like any other generic drive.

When you return to the main Black Box Drive Configuration Menu, you can set each parallel floppy drive to respond as drive 1-8, just like you previously configured your hard drive partitions and serial floppy drives. A column headed "US" shows whether the drive is set to operate in ultraspeed. If ultraspeed is OFF, your parallel drives will read and write at about the same speed as a stock Atari 1050. Use this setting if you want to boot commercial software that doesn't cooperate with ultraspeed.

WHICH DOS?

It should be obvious you'll need a DOS that supports the features your parallel floppy drives offer. If you've already using the Black Box with a hard drive you must be using either MYDOS or Sparta-Dos since no other DOS for the Atari XL/XE supports hard drives. Both are extremely well suited to the FB. If you haven't acquired a hard drive, the above DOSes are still the best choice, although TOPDOS 1.5, MachDOS 3.7 and OSS DOSXL can be used since they support double-sided drives. A caution: these lesser DOSes may have difficulty coping with some of the higher-capacity floppy drives the FB allows you to use.

HOW FAST?

"Get to the good stuff," you say. Okay, here it is. The parallel floppy drives are fast, amazingly fast, sometimes. For maximum speed on disk reads and writes two conditions must be fulfilled: (1) the drive must be set to ultraspeed and (2) the disk must be initialized in the ultraspeed (fast) format.

Thoughtfully, CSS has included a floppy fast formatter as another addition to the configuration menus in the FB-Black Box firmware. Table 2 shows the read and write performance you can expect from the parallel floppy drives when coping with true double density disks. In the chart, parallel floppies are called "PBI" floppies short for "parallel bus interface." I've included figures for a few other drives to compare performance.

Table 2. Floppy Drive Whole Disk SSDD
Read/Write Comparison

Drive	Read Time	Write Time
Indus GT(slow mode)	3'12"	3'46"
HAPPY 1050	1'05"	1'36"
Super Archiver 1050	1'05"	1'03"
PBI floppy(not fast-format)	2'04"	2'04"
PBI floppy(fast format)	0'24"	0'24"

The tests used the copy program included with the "Taskmaster" buit-in sector editor. This copy program does not write empty sectors so I set up the test disk with data in all 720 sectors to avoid skipping sectors during disk writes. A word about the selection of drives: I used what I have. The Indus GT was acquired without information on how to access the "syncromesh" speed and no "Turbospeed" board. While a number of Atari 1050 drives reside with the author, all have been modified to some extent; there's not a stock 1050 in the house.

So why didn't I use one of the fine freeware sector copiers that are available? Most of them like COPYMATE 4.3, COPYMATE 4.4, and MYCOPYR! 2.1, do not recognize floppy drives operating on the parallel bus. CSS is hard at work producing an ultraspeed disk copier to correct this situation. The new copier and an ST/IBM disk reader will be sent to FB owners as soon as they are ready.

COMPATIBILITY

Freeware disk copiers aren't the only compatibility problem with PBI floppy drives. There are a few commercial disks out there, usually heavily protected, that either wont completely load or run from a PBI floppy drive. Quickly running through my library, I found a couple of examples: the EPYX-Lucasfilm Rescue on Fractalus and CSS's own Ultra Menu. Several commercial disks require ultraspeed be disengaged. Ultraspeed can be turned OFF in two ways:from the Black Box Drive Configuration Menu, or boot the disk while pressing the SELECT and OPTION keys.

You'll need to disengage ultraspeed to run M.U.L.E., Run for the Money, and Seven Cities of Gold for example. Finally, disks which require the loading of a "Translator" program, which installs a facsimile of the old 800/400 operating system, will not load from the PBI floppy drive.

FLOPPY BOARD continued

Why? The old operating system did not have the code to support parallel bus accessories; the 800/400 uses those codes locations for other purposes. The solution to these few compatibility problems is simple. Keep at least one floppy drive operating off your serial port to load those "problem" disks.

MORE TO COME

The ad's say the Floppy Board has a version of the Super Archiver built in. How does it operate? For the answer, we'll have to wait for the final version of the Floppy Board manual. Only the preliminary manual is available at present. I can say it doesn't appear the "built in" Super Archiver firmware is enough by itself, to do any of the tasks associated with the CSS Super Archiver 1050. When a Super Archiver 1050 and it's software are operated with PBI drives and the FB, however, the PBI drives can perform most of the Super Archiver copy functions. More sophisticated features like creating a phantom sectors are beyond the ability of the "Super Archiver" PBI drives. Interestingly, the PBI drives also seem to functon as accessory HAPPY Archivers when operating a HAPPY 1050 and HAPPY Archiver software in conjunction with the FB. Bob Puff, President of CSS has promised the final manual will feature a full explanation of the Floppy Board's Super Archiver capabilities and how to use them.

BOTTOM LINE

The Floppy Board is an excellent accessory for the Black Box, especially for those who want to operate high speed, high capacity floppy drives with the Atari 8-bit computers. It's speed really shines when backing up large hard drive partitions onto floppy disks. Operators of small bulletin boards will find it the simplest way, short of acquiring a hard disk drive, to increase the capacity and speed of their boards. The few compatibility problems with commercial software are easily countered by retaining a serial port drive in ones computer system. While the preliminary manual is brief, it's complete and easily understandable. In case of difficulty, the pleasant staff of CSS are always ready to help.

The Floppy Board is available from Computer Software Services (CSS) PO Box17660, Rochester, New York 14617.

PUTER PROGRAM

by Ed Hall

(Fr)Antic, Newsletter of the Alamo Area Atari User Association August 1991

I have two children aged 3 and 4, and whenever they see me at the 'puter, they clamour for "their program." It's only a few lines of BASIC, but it delights them with a noisy, colourful response to their use of keyboard and joystick.

When a key is pressed, the appropriate letter or number is placed on the screen, along with an accompanying sound effect. Pressing the CAPS or INVERSE keys will change the colour of the text; four colours are available.

PUTER PROGRAM continued

If a key is held down, the letter and sound repeat in much the same fashion as on an electric typewriter. If you've ever used one of those ancient beasts, I'm sure you've amused yourself the odd time by making it sound like a machine-gun.

The basis for this part of the program comes from a shortie called "Beginner's Keyboard" by Marty Albers. It appeared in COMPUTE's Third Book of Atari and can be found in lines 210-300. I fiddled around with the sound statements in lines 270-290 to make them more to my liking. You might want to experiment with them too.

To this, I grafted on a joystick routine from a program called "Baby Box" by Joey Latimer, from the January 1986 issue of Family Computing. It's found in lines 70-200. Moving the joystick or pressing the button causes the screen to flash and some random notes to be played.

A word of caution--you should probably keep an eye on the kids while they're at the computer. Sometimes a child will get carried away by the excitement and really start hammering at the keyboard. That way lies a repair bill.

One last thing. I added a short title routine and put my children's names where you see the word "KIDS" in line 40. You might want to do the same.

```
10 GRAPHICS 0:SETCOLOR 2,0,0:POKE 752,1
20 POSITION 10,8:? "'puter program for":GOSUB 310
30 GRAPHICS 2+16:POKE 16,112:POKE 53774,112
40 POSITION 7,4:? #6;"KIDS"
50 GOSUB 310:? #6;CHR$(125)
60 CLOSE #1:OPEN #1,4,0,"K:"
70 R=0:L=0
80 X=0
90 IF STICK(0)<>15 OR STRIG(0)=0 THEN 160
100 IF PEEK(764)<>255 THEN 210
110 IF X=0 THEN 90
120 X=X+1
130 IF X<50 THEN 90
140 SOUND 0,0,0,0
150 POKE 708,25:SETCOLOR 4,0,0:GOTO 80
160 X=1
170 POKE 708,INT(RND(0)*16)*16+4
180 SETCOLOR 4,INT(RND(0)*16),6
190 SOUND 0,52+INT(RND(0)*12)*9,10,6
200 GOTO 90
210 GET #1,I
220 IF R=20 THEN R=0:L=L+1
230 IF L=11 AND I=155 THEN ? #6;CHR$(125):GOTO 70
240 IF L=11 AND R=19 THEN ? #6;CHR$(125):GOTO 70
250 IF I=155 THEN R=0:L=L+1:GOTO 90
260 POSITION R,L:? #6;CHR$(I)
270 SOUND 0,2*I,10,8:SOUND 1,4*I,10,10:GOSUB 320
280 SOUND 0,0,0,0:SOUND 1,0,0,0
290 SOUND 2,5*I,10,8:GOSUB 320:SOUND 2,0,0,0
300 R=R+1:GOTO 90
310 FOR T=1 TO 600:NEXT T:RETURN
320 FOR T=1 TO 25:NEXT T:RETURN
```

HINTS & TIPS.

To start this issue's selection of Hints and Tips, I'd like to give you a few useful POKES.

POKE 580,1 causes a coldstart when (RESET) key is pressed. POKE 580,255 returns the (RESET) key to its normal state.

POKE 65,0 shuts off the beeping you hear during cassette and disk I/O. POKE 65,3 turns it back on again.

POKE 77,129 immediately starts the attract mode.

Send your output to the printer instead of the screen with POKE 838,166:POKE 839,238. Then POKE 838,163:POKE 839,246 sends your output back to the screen.

These instructions let you use filenames with numbers, punctuation marks and lowercase letters with DOS 2.0s. To make the changes permanent, type (H) WRITE NEW DOS FILES when the DOS menu appears. POKE 3818,33:POKE 3822,123:DOS

Typing X=USR(3352) will format your disk from BASIC.

PEEK(53770) will produce a random number between zero and 255.

If you need to enter DOS, but you don't want to loose your program and you don't have a MEM.SAV file, create a MEM.SAV file from BASIC by typing X=USR(5947). NOTE: To return to BASIC, XL owners must press the (RESET) key after the disk stops turning. The 400 and 800 will generate an ERROR 9 message when the MEM.SAV file has been created.

REHABILITATING DISABLED (BREAK) KEY

Include this routine at the beginning of your next autoboot BASIC program for an elegant method of selectively disabling the (BREAK) key.

When your program is booted normally, it will be safe from overly-curious or clumsy fingers. If you wish to modify the program, boot it while pressing the (SELECT) key and the (BREAK) key will re-enabled.

```
1 BRK=1:IF PEEK(53279)=5 THEN BRK=0
2 GOTO 10
5 IF BRK THEN POKE 16,112:POKE 53774,1
12:POKE 580,1
10 REM START OF YOUR PROGRAM
20 REM (USE "GOSUB 5" TO DISABLE (BREAK) KEY)
```

RAMTOP RESET

Next time you alter RAMTOP (location 106) to make room for your own machine language routines, use the line below to make sure RAMTOP is reset to it's original value.

```
POKE 106,PEEK(740):GRAPHICS 0
```

MIRAX FORCE

If you are having problems with this fast action shoot-em-up don't worry, help is at hand. The author, Chris Murray, has built in a cheat mode.

His middle name is Peter and by typing his initials CPM at the title screen you will turn off the collision detection for your craft. You can still shoot the alien craft and destroy the super structure of the mother ship, but they can't get you.

I have used this cheat to get all the way through the game and, believe me, some of the later screens are superb.

Well that's all for this time, I have to say that after 7 issues, I'm beginning to run out of material for this column. It would be a great help if some of you out there could send in some of your own little hints and tips that you would like to pass on to other 8-bit users. It doesn't have to be just programming hints, it could be a tip for one of the many utilities or even a few hints for your favourite games. Why not help me keep this column going. I know it has been of interest to you as many of you have told me in your letters to me. I will use all hints and tips sent in and hopefully with your help, we can keep this column going for a long time yet.

In the Hints & Tips of issue 6 is a small program to slow the scrolling of any program on the screen. Here is a POKE you can do to slow the scrolling.
POKE 622,255:GR.0
POKE 622,0 will return the scrolling back to normal or fast, whatever you want to call it.

This is a tip for users of the DATABASE feature of MINI-OFFICE II. It is of use ONLY to users with more that 64K of RAM, i.e. you need a RAMDISK in order to use this tip.

Anyone who has done a SORT with the MINI-OFFICE II DATABASE will be aware of the painfully slow process. By using the RAMDISK facility, the SORT process can be speeded-up quite considerably.

At first glance this proceedure may seem complicated and cumbersome, but it isn't really.

Here's what I do to speed up the SORT:-

BOOT up the MINI-OFFICE II DATABASE.

From the DATABASE menu select DISC FILE MENU.

Then, LOAD DISC UTILITIES.

From the DATABASE DISC UTILITIES menu, select:-

COPY DATABASE.

Enter drive number (drive 2 in my case)

HINTS & TIPS continued

Select the appropriate Database file to copy.
Input new filename (I use the same filename, but this could be a little dodgy if anything does happen to go wrong).

Destination drive = 8 (probably)

Then from the DATABASE DISC UTILITIES menu, select LOAD DATABASE.

From the DATABASE menu, select DISC FILE MENU.

From the DISC FILE menu, select OPEN DATABASE.

Enter Drive No. (8)
The Database that you just copied from drive 1 or 2 to drive 8 will appear in the window, just press RETURN to OPEN it.

It will be opened immediately.

From the DISC FILE menu, select DATABASE menu.

You can now go about your normal business, adding, deleting records, etc..

When you are finished editing, you can now do your SORT. The SORT will be done on drive 8 i.e. in MEMORY as opposed to disk drive 1 or 2. it really is very fast.

When the SORT is done you will then need to copy your file from drive 8, back to from whence it came (drive 1 or 2).

GOTO the DISC FILE menu and CLOSE DATABASE.

Then LOAD DISC UTILITIES.

As before, COPY DATABASE from drive 8 to whatever.

As I said, it sounds complicated, but once you get used to it, it's no bother, and STILL much quicker that a normal SORT.
Now for the customary disclaimer:-

I assume you will be working on a BACK-UP work disk (you DO have a back-up don't you???) so I won't treat you like an idiot.

If you do use the MINI-OFFICE II DATABASE and have a 130XE or upgraded 800XL etc., then give it a try.

BILL JACKSON.

MARK's GAMES COLUMN

CUTTHROATS by Mark Stinson

Cutthroats, another masterpiece of innovation from the masters of adventure themselves, Infocom, features you as a top class diver searching for sunken treasure. The adventure level is classed as standard, and is second in the 'True Tales of Adventure' series, the other two being Seastalker and Infidel. This series has been my own personal favourite because of its setting in the world of 'high adventure', not that I am totally averse to a moderate sprinkling of horror, magic, mystery and science fiction every now and then!

The packaging is up to Infocom's usual high standard, comprising: a True Tales of adventure magazine, disk, instructions, nautical supply list, and an historical publication by Hardscrabble Harbor Historical Society, entitled 'Four Shipwrecks off Hardscrabble Island'. As always, the adventure has lasting appeal, and a real sense of participation, because of these small touches. I found it especially exciting to read through the shipwrecks book, which shows the believed location of four wrecks off the island. Each wreck is given a brief history, and the unfortunate ships in question are the Sao Vera (1698), HMS Intransigent (1745), Fianna (1887) and SS Leviathan (1916). The True Tales of adventure magazine is also entertaining and amusing, as well as informative.

The story begins in your room at the Red Boar Inn on Hardscrabble Island. There is a knock at the bedroom door which turns out to be a character named Hevlin. Hevlin has dropped in unannounced in an effort to cajole you into looking after his book on shipwrecks. The book has new locations marked for two of the famed wrecks off Hardscrabble Island. After a successful bid, he leaves the notebook in your care, and as you flick through its pages you smile to yourself knowing that so many people have looked for those wrecks without a penny to show for it. The smile is only wiped from your face when you witness Hevlin's murder as he walks the street outside your window. Somehow, you know there is more to this little treasure hunt than one man's over active imagination...

The following morning you awake to find a note from Johnny Red, an associate and central character in the plot. The note asks you to attend a meeting at the Shanty, the islands local bar. You will soon notice that your watch is suffering from old age and abuse, and is completely useless unless wound first thing in the morning. Depending upon how quickly you leave for the Shanty, you may meet the weasel, a thoroughly nasty character who can often be seen looking furtive and up to no good; don't trust him! Once at the Shanty you get to meet the central cast of characters: Johnny Red, the weasel, and Pete the rat (who gained his nickname through his choice of dish while a cook on another ship).

Greetings over, a further rendezvous is decided upon, and after testing the local cuisine you are able to explore the island. The island is quite small, with most of the interesting areas at its north end. At the north end can be found the Wharf, the Red Boar, the lighthouse, McGinty's Salvage, Outfitters International, and the Shanty.

CUTTHROAT continued

To the south of the island is the mariners trust, where you may obtain your cash. By now, you will most probably have met up with McGinty, who is the 'J.R.' of Hardscrabble Island, and owner of the disreputable McGinty's Salvage. My only advice is avoid, and if contact is inevitable keep quiet, McGinty is shrewd and notices everything.

When you finally make your next meeting with the gang, you will be shown one of two objects - a coin or a plate. Examining either of these, in conjunction with your notebook will identify which wreck you are to look for. After an indication of your acceptance, you agree to a further meeting with Johnny. You will also need to withdraw a sum of money before your next meeting. So, having withdrawn your life savings and met again with Johnny, you pay a visit to the Outfitters International. You must now decide which equipment is necessary for the job (depending upon the depth of the wreck); Johnny organizes the ship.

Once all of the necessary equipment is ordered, you can wander about again while waiting for the high tide (check out the tide tables on the back of the equipment list). Timing is very important in Cutthroats, so it is advisable to make a note of certain characters movements during the game. Eventually you arrive at the ship which is docked at the Wharf. You now wait for the arrival of the delivery boy with all your equipment, and after him, the crew. Now is the time for your map reading skills! Before you can set sail, Johnny will need to know the longitude and latitude of the wreck. Once you have informed Johnny of the location, it is advisable (in fact unavoidable) that you get some sleep.

Before long you are woken by Johnny, and realize that this is the moment of truth, now is the time to kit up and dive. If you have bought the wrong equipment, or if you use it incorrectly, you will meet a cold and watery end. It is at this point that the real puzzles are put before you, especially when you try to negotiate the shipwrecks. As I have mentioned, there are two different ships to dive at, each with different puzzles and treasure. I especially enjoyed negotiating the SS Leviathan, on board which, amongst other good puzzles, I had to avoid a live mine which was blocking an exit. Once you have solved all of the ship's puzzles, and retrieved the treasure, you have to negotiate a crew which harbors a cutthroat. Was it all in vain? As there are ten points left to get, you just know there is a solution to this problem, somewhere!

I thoroughly enjoyed cutthroats. Indeed, of the fifteen Infocom adventures I own (only three of which are solved) this is my all time favourite. I have only one complaint, and that is the random selection of the shipwreck to dive on. I spent three quarters of an hour trying to generate the Leviathan version, after completing the Sao Vera!

Availability: Miles Better Software

Price: nine pounds ninety five pence + p&p

Some Hints and Tips

My room is ransacked when I'm out, what can I do?

esu ruoy yek

McGinty prevents me from hiring a ship
peek teiuq dna tel ynnhoJ laed htiw eht rettam

I can't find the wrecks
edutignol dna edutital sgnidaer era no ruoy koob (edisni tnorf revoc)

On the Sao Vera
I break the ladder entering the ship
uoy tnod deen ti

I can't move the bunks
esu eht ekip

The bunks fall back when I move them

egdew meht htiw eht ekip

I can't carry the chest

hsup ti hguorht eht oleh dna hcatta ti ot eht epor

How do I get up the ladder?

uoy tnod, hsup eht larrab ot eht htron, dnats no ti dna tuc epor (htiw drows morf noteleks) - bmilc epor

Whats the machine for?

sti a latem rotceted!

On the SS Leviathan

How should I get into the dining room?

hguorht eht eloh erehw eht enim si

How do I stop the mine from exploding?
tup tengam neewteb sekips, nrut ti no, pord ti

How do I squeeze through the gap?

dloh knat elihw gnimmiws hguorht

How do I open the safe?

llird ti

How do I stop the case from filling with water?

ekat ti ot eht ria tekcop, llird ti ot niard retaw, riaper ti htiw yttup

HOW DO I PREVENT MY MURDER WHEN FINISHING THE GAME?

yap a tisiv ot s'ytniGcM retfa uoy evah thguob ruoy seilppus ta eht rettiftuo, tub ekam erus eh si tuo. retne raer wodniw, teg epolevne. edih ti rednu ruoy sserttam dna evig ti ot ynnhoJ ta eht evid noitacol.

IS ANYBODY LISTENING ?

by Jim Cutler 25th November 1993

Or the arcane science of talking to your computer

I use a Bog standard ATARI 130XE for wordprocessing but also have one of the excellent ATARI PORTFOLIO 16bit MS-Dos notebook computers which has an excellent built in Text Editor. I find this useful to produce ASCII text files, say on holiday, and port them across to the 130XE. The items to do this are not vastly expensive. You need

A) A 9 pin socket to 25 pin D RS232 plug. I use the DATATARI interface

B) A Null Modem Maplin YP85G price £5.95

C) RS232 male to female 25 way Maplin JC13P price £6.95

D) RS232 25 way D type to 9 pin to suit you second computer.

The 9 pin fits into a serial port on each system.

You need a Comms programme to manage the operation. I use XTERM2 Comm on the Portfolio and MINI OFFICE 2 Comms module on the 8 bit.

Transfer at 1200/1200 bps of ASCII is simple and fast.

Prices are from MAPLIN'S 1994 catalogue.

For the more technically inclined see the file PORT85.txt.

PORT85 -> 8-BIT CONNECTION

In order to connect your 8-bit to the new PORTFOLIO computer, you have to know three things;

1. You have to have an RS232 interface connected to your 8-bit; the 850, P:R: CONNECTION, or MIO devices all work well for the purpose.

2. You will need a null modem cable. This is a cable which connects two computers without the need for modems (cutting out the middle man, so to speak).

3. You will need to make the null modem cable (or have a friend do it if you subscribe to the ancient Code of the Programmer); "Hey, I don't do that, it's a hardware problem!"

PART ONE: THE PROBLEM

The PORTFOLIO has file transfer capability... through the parallel interface. I don't know of any 8-bit application to redirect communication software into the parallel port, so ignore the File Transfer section in the SETUP application.

The RS232 ports on 8-bit computer interfaces are not compatible with "standard" RS232 DB-9 cables, (having been developed a few years before IBM stuck their foot in the PC door); pinouts are as shown below.

PORT85 continued

8-bit	STANDARD
1 DTR	1 CD
2 CD	2 RD
3 TD	3 TD
4 RD	4 DTR
5 GND	5 GND
6 DSR	6 DSR
7 RTS	7 RTS
8 CTS	8 CTS
9 (none)	9 RI

Fortunately, PORTFOLIO's Serial Interface does conform to this standard, so that modems and IBM AT cables can be used with it.

One other problem; the diagrams in the Serial Interface manual are wired wrong. If you intend to make your own standard cables, refer to this service for the file detailing proper connections. (NOTE: this file may be found in the ST sections of this service; for CIS, GO ATARIPRO; for GEnie, type ST.)

PART TWO: THE SOLUTION

Well, Part One ought to have scared the willies out of all but the most determined Solder-Jockies, so we can now proceed with our interface.

1. Get two (2) DB-9 "D" style connectors; one Male, one Female. (NOTE: if you don't want to go through the hassles of soldering, pick up the Radio Shack crimp-style connectors (CAT #276-1427 & 276-1428, respectively); with these connectors, all you do is insert the wires and crimp the pins closed.)

You will also need hoods for your connections; I used the metalized hoods (276-1513) for my cable, as they offer the shielding ability of metal with the light weight of plastic.

For the cable, I recommend (278-775) double-shielded cable, especially for the MIO user; the PBI bus is flat-out full of RFI, and the chance of data corruption from that monster makes the extra cost worthwile.

Connections are as follows;

```
8-bit (MALE)  PORTFOLIO (FEMALE)
+--1 DTR            4 DTR--+
+--2 CD             6 DSR--+
+--6 DSR            1 CD (N/C)
   5 GND------------5 GND
   3 TD-------------2 RD
   4 RD-------------3 TD
+--7 RTS            7 RTS--+
+--8 CTS            8 CTS--+
   9 (N/C)          9 RI (N/C)
```

Cable shield attached to PF hood ONLY.

Double check your connections before connecting to either computer.

PORT85 continued

To test your interface, connect the cables to their respective interfaces and fire up your favorite 8-bit term software (I've used 850 EXPRESS! and BOBTERM with equal results), and set the terminal for 300 baud, half duplex, ATARI. The ATARI mode will be important once you start binary file transfer.

For PORTFOLIO, go to the RS232 option in the SETUP menu, and set 300 baud, no parity, 1 stop bit, 8 data bits, and initialize.

Force the TERM mode on your 8-bit. At the c) prompt, enter: COPY CON AUX. Type something in to your Portfolio, press ENTER: your message from PORTFOLIO should echo on your 8-bit monitor. If not, check the term program settings, connections, and cable pinouts.

CONGRATULATIONS! You now have a handheld computer that "talks" with it's older brother.

The advantages? You can use your full-screen 8-bit for communications and D/L to your palmtop. You can fill your Portfolio with programs without having to borrow somebody's IBM. And, best of all, you can compute respectfully during the day, and still blast away at your favourite arcade game at night.

THE PARALLEL BUS REVEALED.

100.000 BYTES PER SECOND
Part II of a four part series
By Earl Rice.

Last month we looked at the general operation of the Parallel Bus Interface (PBI) emphasizing software concepts. This month, we'll look at the hardware concepts involved in making an external device work via parallel bus.

Figure 1 shows the basic requirements for a simple serial I/O function such as an RS-232 board. The serial I/O device can be an integrated circuit that looks like a set of registers to the computer. The decode logic selects the device when the assigned PBI addresses are presented on the address bus.

Beyond that, the computer needs to be able to read and write data to the device and respond to it's requests for service. That's what the Read/Write and interrupt request lines are for.

When the computer wants to talk to a parallel bus peripheral, it enables the decode logic with a signal called External Enable. The decode logic decides whether the address on the bus is for the device or for the 2K ROM.

If it's for the ROM, the decode logic returns a Math Pack Disable signal to the computer so that internal ROM doesn't contend with the 2K ROM for the data bus.

PARALLEL BUS continued

Figure 2 shows the approximate timing of the External Enable (EXTENB) and Math Pack Disable (MPD) signals.

Figure 3 is a more detailed schematic of the hardware example. This is still not the complete design, but it serves to illustrate the concepts some more.

The decode logic does several things:

* Decodes the $D8XX--$DFXX block to enable the 2K ROM.
* Decodes the D1XX block for device registers.
* Decodes the address $XXFF so the computer can select the Device Enable latch at $D1FF. This latch represents the select bit in the Device Select register (DEVSEL).

In figure 3, the latch is tied to the data 0 line (for device 0). But it could be tied to another line to make the device respond to a different ID number--such as bit 3 for device 3. For simplicity, the latch is a write-only bit. When we expand to a full design, we'll see how to make a readable register. We would need to do that to make the peripheral available to an interrupt service routine.

The Device Enable latch must be set to allow the computer to address the 2K ROM or the device registers. It's output is also gated with the 2K ROM select line to send the MPD signal to the computer. The Device Enable latch is set by writing a 1 to $D1FF. It is reset by writing a 0 to $D1FF.

SUMMARY

The basic concept isn't hard. The external hardware is enabled by EXTENB. It must decode the 2K ROM space, device registers, and Device Select register. When the 2K ROM is selected, it must return a Math Pack Disable signal to the Computer. If the device is interrupt driven, it must supply an Interrupt Register(IRQ) to let the computer know it wants to be serviced. In that event it will also need to supply an IRQ ID number to the computer.

Simple? Sure is...At least at the conceptual level. It gets a little more involved when we design the decoder and have to deal with timing. We'll do that in next issue. In the meantime, here's some additional information to help you understand the diagrams:

READING THE DIAGRAMS

Some of the signal names in Figure 3 have a bar over them. That means the signal's active state is low (binary 0). The little circles on the signal lines next to some of the symbols mean the same thing. A dot at the intersection of two lines means they are connected. If they cross without a dot, they aren't connected. The wide arrows indicate multiple signal lines. The numbers inside the arrows tell you which signals are involved. For example, the arrow with A0--A11 means the address lines for address bit 0 through address bit 11 all go to the 2K ROM.

PARALLEL BUS continued

Only address bits A0--A7 go to the $XXFF decoder and the device registers, however.

Figure 4 shows the pin numbers for the PBI signals. It also shows how the pins are numbered on the printed circuit connecter at the back of your 800XL cmputer. Imagine you are looking into the open connector slot.

CAUTION: Unplug your computer before removing the cover over the slot. Be very careful not to short the connector pins. Before touching a connector, be sure to discharge any static charge you may have built up by touching a grounded conducting object (such as a cold water pipe). STATIC CHAGES CAN DESTROY INTEGRATED CIRCUITS! BE CAREFUL!

In next issue, we'll look at a serious design for a sort of baby 850 interface--a serial I/O device usable for driving modems and such.

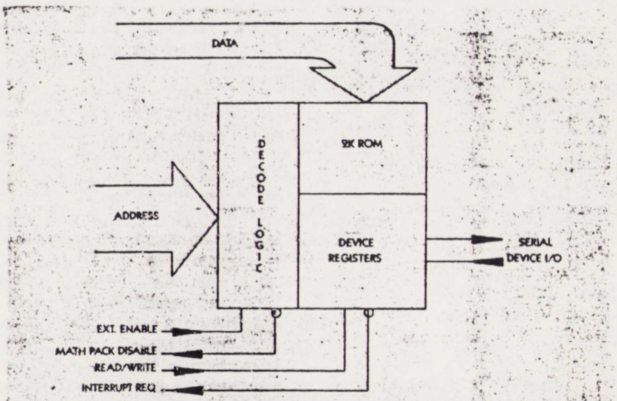


Figure 1.
Serial Interface Block Diagram

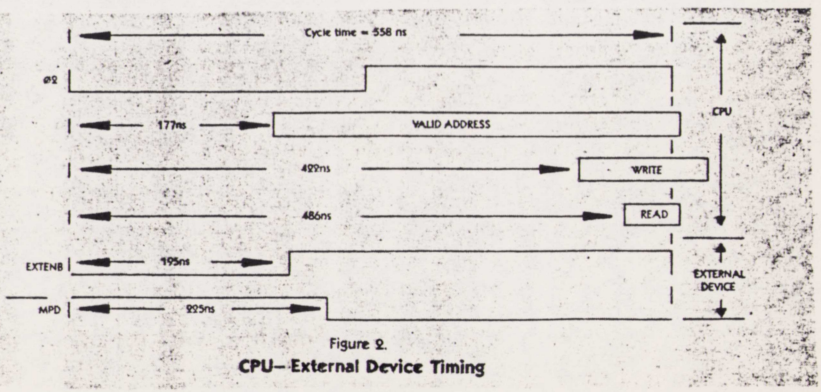


Figure 2.
CPU--External Device Timing

PARALLEL BUS continued

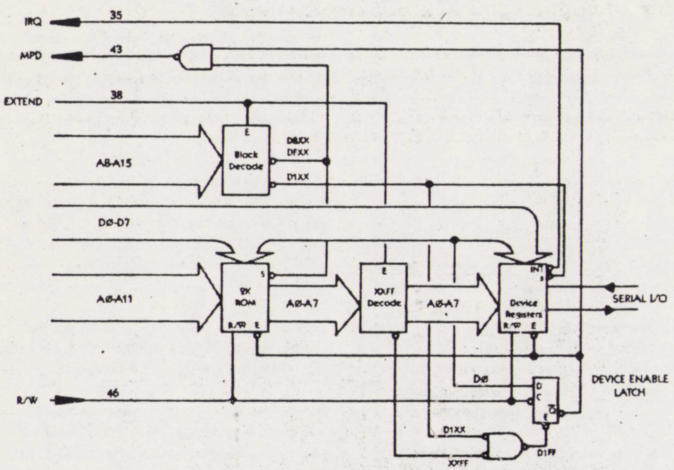


Figure 3.
Simplified Serial Interface Schematic

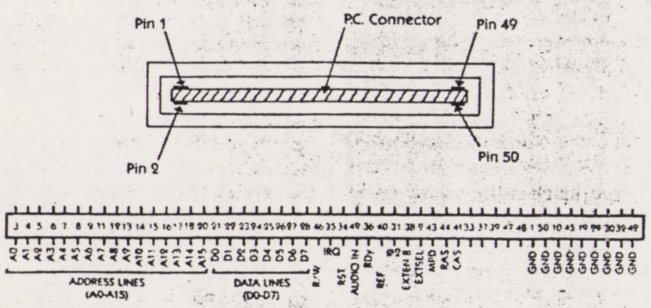


Figure 4.
Parallel Bus Pinout

LETTER SECTION

by John Dickerson, DACE

Greetings to members of the Tyne & Wear club from DACE! We are exchanging newsletters and DOM's (disks of the month) with you and have been excited to find a club in England. We are in contact with clubs in a number of other countries as well.

We are located in Concord, California, and together with the SLCC and ABBACUS serve the Atari users of the San Francisco Bay Area. I am a member of all three clubs and the Treasurer and 8-bit newsletter exchange contact for DACE. DACE is over 10 years old and has hardly missed an 8-bit DOM in all that time, but the majority of our membership is now ST oriented.

Several exciting things are going on in the US. One is that Atari Classics Magazine is starting its second year of existence. This is an all 8-bit magazine and I have found it to be excellent. It is run by volunteers and their goal has been to support the few of us 8-bitters left with only 500 subscriptions. It would be great to see more international participation in this endeavor. If you wish more info, write Atari Classics, 179 Sproul Road, Route 352, Frazer, PA 19355 USA.

Another successful venture has been the International Atari Network (IAN for short). This is a linking of a message base by BBS Pro bulletin boards. All messages are in English, only the 8-bit is discussed, and the turn around rate is about a week. Presently Holland and Germany are the only European countries in the net and it would be exciting to have England also. Setting this up would be expensive and time consuming, but it would be a great service to the international 8-bit community. If anyone is interested, drop me a note and I will put you in contact with someone who can help you get started. My address (and DACE's also) is PO Box 673, Diablo, CA 94528 USA.

My favorite new program is VTEX. I am sending your club the latest release, Version 1.2, for distribution. This is a .DOC and .TXT reader program that makes reading these files on screen a pleasure rather than a chore. Load it up and press "?" for the help menu. Press "R" to read a disk inventory and "L" to load a file. You will want to look at the documentation included with the program. VTEX seems to work with all DOS's and all machines and drives. This is a shareware program so send the author a few bucks and your comments if you like it. We may end up with an even better version!

Dear T.W.A.U.G.Team,

Just to congratulate you on reaching your first birthday issue. Many thanks to your team of contributors, reviewers etc. The magazine has a good selection of programs, games and reviews and just goes to prove that the ATARI Classic 8 Bit is NOT just a Games Machine.

I am enclosing my subscription for the next 6 issues

BEST WISHES Tommy Wood

CONGRATULATIONS TO TWAUG!!

I just felt I would like to write and congratulate the TWAUG team on 6 excellent issues of the Newsletter. That means it's 1 year old, HAPPY BIRTHDAY !!.

It doesn't seem 5 minutes since I placed an add in Micro Mart asking for Atari contacts and David Ewens wrote to me telling me about the newsletter and disk which the team had just started producing. I sent off my subscription, but never expected to receive anything of the quality that the mag and disk turned out to be!.

I was truely amazed at the work and many long hours that David, Max and John must put into producing each issue. Anyone who has ever tried their hand at DTP knows how time consuming it is, and compiling and duplicating the disks, printing labels, mailing etc. How do these guys do it and still have time to eat and sleep between issues?.

Lets give them all a pat on the back for their dedication and hard work supporting us Atari Classic users. Without people like them the best little 8-bit would be dead!.

Here's to many, Many, more years of the Newsletter and TWAUG!!. Also a special thanks David without whose letter I wouldn't have known of the mags existance.

Tony Bingham, Lincs.

Thank you Tony for the very nice letter. Over the last 12 months it's letters like yours that we have received from many of our subscribers that have helped to keep us going.

BUY and SALE

FOR SALE.

1027 printer with PSU, good working order £25 + £5 P&P. 1027 printer with PSU not working, good for spares £10 + £5 P&P. 2 410 tape decks not working, no PSUs. Good for spares.

WANTED.

Light Gun and any ROMS for use with it, also any other ROMS, CX85 pad, Touch Tablet and Atari Artist ROM. Contact, Mr. D. Heard-white, 34 Mayfield Cre., Brighton, BN1 8HQ.

XF-551 UPGRADE

reviewed by John Dickerson

At the last DACE meeting here in California I showed off my new toy. It's the XF-551 dual drive upgrade from CSS. I shipped them my clunky stock XF-551 drive and they upgraded it and added a 3.5" drive as drive #2 for $99 US plus my shipping costs.

I still have a lot to learn about all the neat stuff this combo will do. So far I have figured out that the 5.25" drive now works in true double density. This means I can use it for my Synfile data disks. It still won't format the second side of a disk as a single side, but you can get a gadget from CSS which fixes this problem also for about $40. I just first format the second side on a 1050 if the problem comes up, and the XF-551 has no problem writing to the second side then.

It is important to figure out what DOS's to use. I like SuperDos best and have used it exclusively in the past to take the most advantage from my XF-551. It works great on the upgraded 5.25" drive but apparently sees the 3.5" drive as another 5.25". It formats both drives in double sided double density which is over 2800 single sectors. Well, the 3.5" drive should have twice as much space.

So, I dragged out the instruction manuals. SuperDos does not seem to be configurable, or at least I couldn't figure out how to do it, but I knew that MyDos is. I knew MyDos is good for upgraded drives and ramdisks, but I've never used it much because it gets buggy with enhanced density. So I fiddled with MyDos until I got it to format the 3.5" drive to its full extent.

You may well be thinking about whether or not such a setup would really be of much use. The only advantage of a 3.5" drive is that you can store twice as much as you can on a 5.25". Well, so far I have found several uses for this feature.

Firstly, BobTerm works very well with this MyDos configuration and I can batch download to my 3.5" with much less hassle then before. With over 5600 sectors(2800+ double density sectors), I do not have to worry nearly so much about how much space I have left on a disk.

Secondly, the 16-bit world occasionally gives us text files that are just too big for a regular disk. I have downloaded one of these to my 3.5" and have been able to read it with VTEX with no problems.

Well, that's what I have learned so far, but I am sure there is much left to discover. SpartaDos (which I haven't figured out yet) can be fixed up to work with this setup and I am wondering about Textpro (which I also don't know). I have a SpartaDos version of AtariWriter which works well except I haven't figured yet how to get it to access all of the 3.5" disk. I hear there is a MyDos patch version available and I am looking for it.

If you have an XF-551 I would recommend this upgrade. Even if you don't want the 3.5" drive add on, send in your XF-551 just to get it fixed up. I bought mine not long after they came out for about $200. It was not long before it was in for service because of the flimsy connections to the IO cable sockets. Apparently many of the earlier XF-551's had this problem. And of course Atari never came up with a reasonable DOS, you can't format the second side, there was no double density, and there was no software, commercial or PD that was ever developed to take advantage of it's unique properties. Mine went up into the attic until I got SuperDos Ver.5. But now since I got it fixed, I find I'm using it as my primary drive! Atari would have been better off to just upgrade the 1050 to US Doubler capacity and buy SuperDos or MyDos for it.

CSS's address is 983 Pixley Road, Rochester, NY 14624, Ph. 716-429-5639.

Now I will tell you about my next project. I am contemplating entering the modern world and getting a memory upgrade! I have never in the past considered such a step for long, because there was no software that I used that would take advantage of such a feature. But I have recently heard that Synfile may utilize the upgrades, and that is very exciting to me. If anyone knows anything about this, I would be delighted to hear from you. I am also wondering about whether to get a 1 or 4 meg upgrade. What's a good use for 4 megs of RAM? My mailing address is PO Box 673, Diablo, CA 94528 USA.

COLOUR.BAS

Here is a nice little type in BASIC program by Ralph Bradley (1989) to do some colour test.
Use the cursor keys to change colour (- and =), and luminance (+ and *).

```
10 REM D:COLOUR.BAS
20 REM (C) R. Bradley 1989.
30 REM ↑ ↓ COLOUR   ← → LUM.
40 REM
50 GRAPHICS 0:POKE 752,1
60 POSITION 12,10:? "COLOUR TEST"
70 DL=PEEK(560)+256*PEEK(561)
80 POKE DL+28,7:A=0:B=0:GOTO 190
90 POKE 764,255
100 IF PEEK(764)=255 THEN 100
110 C=PEEK(764):IF C=14 THEN A=A+1
120 IF C=15 THEN A=A-1
130 IF C=6 THEN B=B-1
140 IF C=7 THEN B=B+1
150 IF A>15 THEN A=0
160 IF A<0 THEN A=15
170 IF B>15 THEN B=0
180 IF B<0 THEN B=15
190 SETCOLOR 2,A,B:POSITION 0,23
200 ? " COLOUR= ";A;"LUM= ";B;"  ";
210 GOTO 90
```

OPERATION BLOOD (LIGHT GUN)

Review by Mark Fenwick

It's time to dig out your trusty Light Gun, wipe away the dust and cobwebs and plug into your 8-Bit for some real Light Gun action. For Operation Blood, Light Gun version is here!

Operation Blood is clearly the best selling 8-Bit game for the nineties. With this new version utilising the Light Gun, it should easily follow suit.

Operation Blood is a fast and furious shoot 'em up in which you must shoot at a given amount of foot soldiers, armoured vehicles and helicopters for each level. The graphics are very well detailed (no Mr Blobby graphics here!) down to facial features on the enemies faces. Each level is set out differently to it's particular scene as you progress through the levels, for instance the jungle level is set out as a, yep you guessed it, a Jungle. The background graphics give a good sense of depth as are the soldiers coming at you in degrees of size. The action is very fast, with the battlefield full of enemies giving you what for at any one time.

You begin play with a limited amount of bullets and grenades (which soon depreciate) and a bar showing health which depreciates with each shot that you take. Don't despair though, as there are random icons which appear at the bottom of the screen to replenish health, ammo, grenades as well as a rapid fire symbol. These power up icons are easily collected simply via shooting at them. Your amount of grenades, rounds of ammo and number of kills is displayed in a very small portion of the lower part of the screen, leaving plenty of room for the play area.

As with the original version, shooting the enemy is done by placing the cross haired sight over your enemy and shooting. With this Light Gun version you must move the sight by pointing your gun at the enemy. I was quite surprised to see the sight remain on the screen all the time, so you don't have to fire first to check you've lined your gun sight up properly. This is much better as with other Light Gun games you tend to be sat squinting with one eye the other shut trying to see where you've shot, which given the accuracy of most guns this can prove very frustrating! None of that here simply hold your Light Gun in your hand and as long as it's pointing to the screen somewhere, you can just follow the sight on the screen as you move the gun. With a bit of practice you'll soon have your name on the High Score table!

When you lose all your health and get a score to qualify for the high score table, you'll see the familiar gun sight above the letters A-Z followed by a couple of icons. Don't try to shoot at your letters with the gun, as this won't work! You must keep a joystick plugged into port 1, (This is not stated in the instructions) this is no great detrement as your Light Gun is plugged in to port 2 anyway, move the joystick to your desired letters pressing fire after each one, until complete, your name or rude comment will now be saved to disk. One thing I did notice was the increase in values on the high score table so you may need a little practice in getting there.

How does this Light Gun version fair against the original ? In my opinion it wins hands down as it seems faster as you don't have to drag the gun sight around the screen, just slight movement of the hand and your sight is at the other end of the screen. Playabiltity is much more improved by this and holding that gun in your hand does add that bit of realism. Just one thing missing though...Blood! Saddistic aren't I ?

If you've got the joystick version then you'll love this as you'll see more of the game. If you haven't a Light Gun, then beg, borrow or steal one, no, don't do that, if you do get stuck then Derek Fern now has some and with an excellent game like this to use it with it's definately worth a buy. Operation Blood, Light gun version are both available from Derek Fern at a real good price of £5.95. Special thanks must go to Derek for his help in the development of this version, not forgetting the ANG lads who wrote it of course!

SPECIAL FORCES

Review by Mark Fenwick

At last it's here, what we've all being waiting for, the sequal to the best ever shoot'em up game for the 8-Bit, Operation Blood II (Special Forces). With most sequals be it film or game, you always lose the originality of the first version, not here... Different scenarios, better graphics, sound, gameplay and believe it or not, harder.

When you first boot up Operation Blood II a title screen appears, showing title, credits and a scroller giving game controls. Accompanying this screen is a nifty looking tank which drives left to right occasionally turning to shoot a shell at you, a good piece of music accompanies the title screen. From here you press fire to enter the first level.

The screen breakdown is mainly playing area with the bottom part of the screen showing status. The staus bar shows current lives, ammo, mortar shells and hits taken. Icons represent these as well as number of foot soldiers, planes, boats and jeeps to be destroyed for each level.

I was pleased to see that there was more than one life to the game, in fact 3, however I soon found out why. The number of enemies in their various shapes to destroy is quite unreal, on first sight my words were "What the..." you can guess the rest. Obviously with a greater amount of enemies to destroy this greatly adds to the difficulty factor.

The foot soldiers follow the original with there rolls and knife throwing. An added bare chested Rambo-type who pops his cheesy head up from time to time, armed with a machine gun. The uniforms are different as are the vehicles, a jeep for instance drives along with a soldier in it shooting at you like a man possessed. The helicopter has gone, but in its place is a plane which hovers shooting countless rounds at you. Another addition is a boat, which won't be seen until level 2, this too has soldiers shooting at you.

SPECIAL FORCES continued

The action is fast and furious with countless soldiers, along with the jeeps, planes and boats all on the screen at once, action can become mind-blowing.

Power-ups are plentyful, just as well really! While contending with killing everything in sight, power-ups fall from the sky at random with little parachutes attached to them. These must be shot at before they hit the ground, as once they've fell to the floor they disappear after a couple of seconds. These power-ups range from ammo, shells, health and don't quote me but I think I may have shot an extra life?

For the levels, well, given the amount of lives factor it's still a toughy to say the least. I have managed to get to the end of level 2, which is a real toughy! You're in a marsh land, with a river in the distance, you've just got used to blowing the planes and jeeps, then comes along a boat too! Those lives soon go. Each time you take a hit an icon appears on the hits bar at the base of the screen, once this hits passed a certain level a warning bleeper sounds out, basically just to let you know you've shit, sorry shot it. The high score table's still there, so don't worry about losing that terrific score.

Overall this is an excellent sequal to a game which was thought could not be improved upon. There's a hell of a lot more action and alot more to shoot at. One good thing is that although the artillary quota has been greatly advanced, they do in fact, take less shots to destroy, don't think that this makes the game any easier though. A couple of tips for you are, on level 2 wait until the boat is in line with the plane before firing a shell, that way you'll get both with one shot! Don't run out of ammo, as each new life runs off of the previous lives status, so what ever you have left after your first life ammo, shells, kills are all the same.

I purcha-ed my copy from A.N.G. at the show for a mere £4.00, I tried a bit of arm twisting for a cheat for Op.blood from the A.N.G. crew, unfortuneatly they wouldn't part with it. He did say there was one though, so keep trying! If you haven't got Operation Blood II then I suggest you get it, even if you have the first version. I just hope there'll be enough sales to warrant a Light Gun version, what d'ya think Derek? Operation Blood II is currently available from Micro Discount for £5.95, and worth every penny!

ZONG MAGAZINE.

Review by Mark Fenwick

It was good to see A.N.G. and KE-SOFT, both selling their latest software products at this year's annual AMS 7 show. There were plenty of bargains to be had from both in new and old software. One such production, ZONG disk mag from none other than KE-SOFT.

ZONG disk magazine has been around for some time in Germany, since 1989 the magazine was produced solely for the German market, unless you speak/read German, you were out of luck. KE-SOFT realised there may be a possible market over here in England for the mag. So here it is, a 100% translated English version of the disk mag.

ZONG MAGAZINE continued

The magazine is presented very well, an A4 magazine in mono, not forgetting the issue disk on the inside cover. The magazine is set out over 24 pages and there's no wasted space here at all. The idea seems to be to have covered all possible aspects of past commercial magazines. The mag comes in a green cover with screen shots of the disk games on the front and a list of features, plus an icon of a soon to be famous character from the excellent new game "Brundles". The back cover shows the "Brundles" chap somewhat larger as an advert, along with KE-SOFT's correspondence address.

As I turned the page I soon realised the work and effort put in to the mag. The first few pages cover reviews of the very latest games from none other than ? KE-SOFT, with the exception of two classic Atari Roms, each review takes up about half a page, with a whole page dedicated to a review of "Brundles". From the reviews we come to what cannot be missed from any mag, Hints 'n' Tips. Not quite what I was expecting though, better, as most of the games are very recent, and the methods used aren't seen too often. The majority of titles need the use of the 'Freezer' while the others use diskmonitor or equivelant. Each cheat shows what address needs to be set to give number of lives you want, unlimited lives, invulnerable and unlimited time. I don't want to give too much away, but if you're stuck on "Adax" you should quite easily complete it with the cheats here! Easier alterations for older games such as Quasimodo and Treesurgeon to name a couple are done with disky, sector and byte to alter given with each. There's also cheats and maps to various classic adventures, so adventurers out there you can stop banging your head against the wall!

Next is a review section of PD software, with each title well described. A workshop section of seven pages covers many aspects of programming and problems you may encounter. The last two pages are dedicated to the explanation of the programs on the enclosed disk. The A side has a turbo basic autorun menu with seven different programs. These are loaded via a joystick run menu, and cover a varied range from logic to levels games. Each game is well described and the programs themselves are very good too. What about the B side I hear you say, well, it can only be described as Shit! that's got you thinking! No, the title is in fact called "SHIT" and is no baring on the quality of the game. The story of this game is that a fly was eaten up by a fat woman (Why a fat woman would want to eat a measly fly, I don't know!) however, the fly lived so it's down to you to guide the fly out of the stomach to the ex-shit! The graphics are hi-res with a degreeing level of difficulty as you progress, control is good too, will you make it to the shitty end.

Overall I think ZONG is great value for money, every possible topic has been covered. The mag itself is professionally set out, with no wasted space anywhere, giving excellent value for money. With a double sided disk full of stuff too, a price of £4.00 is no great price to pay, when you consider the B side game "Shit". If you were at the show you'll have noticed A.N.G. selling this for £4.00, what more can I say. My only criticism is the lack of a news section to let us know how the German 8-Bit is surviving.

ZONG MAG. continued

I'd just like to point out that although KE-SOFT and A.N.G. were at the show, after speaking to both parties, it's doubtful if they'll be there next year. When you consider the costs of getting here for a start, to try and make that back in a day is no easy task. Take this first trial issue of the ZONG mag in English, when speaking to Kemal Ezcan of KE-SOFT asking if further English ZONG's will be vailable, he stated that he must sell at least 200 to make it worthwhile. When I purchased copies for myself and my good friend David Ewens, he told me he'd only sold 20 copies. A bit of a let down indeed, I cannot urge you enough to buy a copy of this excellent mag, I know there are loads of mags already out there but with the value for money and first hand news of new releases and developments it's got to be a good bet. So if there's anyone out there who didn't get an issue, why not write to KE-SOFT and try issue 1 before saying not another disk mag. Write to Kemal at:-

KE-SOFT
KEMAL EZCAN
FRANKENSTRASSE 24
63477 MAINTAL
GERMANY

FIVE - LINER

These FIVE - LINERS are taken from the Old Atari User Magazine.

REFLECT by Paul Farnfield.

This is a rather neat demonstration of one of Atari's least used features, the hardware reflect facility. The program is written in Basic, but there is a small machine code interrupt service routine which is used to turn the reflect register on during the lower portion of the screen only. This program could have been written in three lines, but the five line limit has allowed a much clearer approach.

Line 10 sets up the screen mode and colour, and the start of the display list.

Line 20 loads the machine code routine into Page 6.

Line 30 tells the operating system where to find the interrupt routine. it also enables the DLI's and sets up the custom Display List.

Line 40 simply positions and prints out the text.

Line 50 just loops around to stop the display from being cleared.

```
10 GRAPHICS 18:POKE 710,0:D=PEEK(560)+256*PEEK(561)
20 FOR W=1 TO 11:READ B:POKE 1535+W,B:NEXT W:DATA 7
2,169,4,141,10,212,141,1,212,104,64
30 POKE 512,0:POKE 513,6:POKE 54286,192:POKE D+6,2:
POKE D+12,135:POKE D+13,6
40 POSITION 15,1:? #6;"DLI REFLECT":POSITION 2,8:?
#6;"BY PAUL FARNFIELD":POSITION 2,9:? #6;"By PAUL F
ARNFIELD"
50 GOTO 50
```

FIVE - LINER

CURSOR HIGHLIGHTER by Tim Baily

Have you ever been de-buging a program or editing a screen full of text and wished the cursor was a little bit easier to see,or even wished you could find out which line it is on?

Well this program solves both problems. It is written in machine code and resides in Page 6 of memory. Its function is to highlight the line that the cursor is currently on. It also flashes the cursor, and any inverse video characters that are on screen.

The program sets up a display list interrupt on one of the blank lines at the begining of the display list. The program calculates where the cursor is with respect to the scan lines of the TV display by checking VCOUNT (54283, the scan line counter), and at the appropriate time changes the colour of that line.

Next it delays long enough to cover eight scan lines-equal to one graphic mode 0 line-and then restores the original colour and returns from the interrupt. The flashing cursor is operated from the vertical blank interrupt, which operates 50 times a second.

If you wish to save or load programs while this routine is running first Poke 54286,64 to turn the display list interrupt off. If you do not, some errors will result in the serial port input/output routines.

On completion of SAVE/LOAD operations POKE 54286,192 to restore program operation. Note the cursor will still flash, but the coloured line will disappear when POKE 54286,64 is used.

After the program is typed in save it before running it. If all is well you will see a dark red line across the screen on the same line as the cursor and the cursor will be flashing. You may type NEW to clear the program memory as the routine sits in Page 6, save from Basic. If you press System Reset or, call a new graphics mode, the line will dissapear. You may restore the function of the routine by typing X=USR(1536)

```
1 FOR I=1536 TO 1536+128:READ A:POKE I,A:NEXT I:X=U
SR(1536):REM LINE HIGHLIGHTER T.G.BAILY 1986
2 DATA 104,169,46,141,0,2,169,6,141,1,2,173,48,2,14
1,32,6,141,35,6,173,49,2,141,33,6,141,36,6,169,128,
13,32,6,141
3 DATA 32,156,169,192,141,14,212,32,83,6,96,72,138,
72,24,165,84,105,4,42,42,162,20,205,11,212,208,251,
142,24,208
4 DATA 105,2,205,11,212,16,251,173,198,2,141,24,208
,104,170,104,64,169,7,160,92,162,6,76,92,228,206,12
8,6,48,3,76
5 DATA 98,228,173,243,2,201,1,208,13,169,2,141,243,
2,169,12,141,128,6,76,98,228,169,1,141,243,2,76,112
,6,11
```

CRACKING THE CODE

by Keith Mayhew

Re-printed by M. Gerum

This article first appeared in "The UK ATARI Computer Owners Club" later renamed "MONITOR"

PART 7

This time we conclude our study of the indexed-indirect modes and find out why they are useful! Having done that we will have covered all the 6502's instructions and addressing modes and will be concerned with using the machine's resources from assembly language, such as controlling the graphics and sound hardware. The operating system will also be covered to see how we can perform input/output operations; we will also be studying the use of interrupts and how the operating system employs them for a lot of its work.

THE NEED FOR POINTERS

To recap from the last issue, the two indexed-indirect modes we studied accessed their data through two-byte pointers stored in page zero of memory. The pre-indexed used the X register by adding it to the given address to create the actual address of the pointer which was then used indirectly, see figure 1. Post-indexed used the Y register and indirected through the pointer first to give the base address of the area and then added on the Y register to give the final address, see figure 2. The key element to these addressing modes is, of course, their use of a pointer in accessing a byte of data; it is important that you understand their operation clearly as they are used very often when programming.

The first reason why a pointer can be used, is to overcome the limitation of the 6502's absolute-indexed modes. As the index registers are 8-bits long, only 256 bytes can be accessed from any given, absolute, base address; the solution is to save the base address in a page-zero pointer and then use an indexed-indirect mode. The idea here is to change the pointer's value to access any byte we choose. We could use either the pre-indexed or post-indexed addressing mode, the following shows the use of pre-indexed mode to move a table of values:

```
      LDX #0          ;Keep zero
                      ;Index.
LOOP  LDA (POINT1,X)  ;Load a byte
      STA (POINT2,X)  ;Save it.
      INC POINT1      ;Inc.low pointer.
      BNE SKIP1       ;Skip if not zero.
      INC POINT1      ;Else inc. high byte.
SKIP1 INC POINT2      ;Same for second pointer.
      BNE SKIP2       ;Pointer.
      INC POINT2+1
SKIP2 /   /
      /   /           ;Perform.
      /   /           ;some test.
      /   /
     "branch to"LOOP;Go back
                      ;if more
                      ;to copy.
```

POINT1 and POINT2 are assumed to be in page zero and to have been initialised to point to the start of two areas of memory, or tables. The X register is kept at zero, otherwise we wouldn't be accessing the same pointer each time! Each time around the loop the 2 two-byte pointers are incremented by one and then some test is performed followed by a branch back to the start if more is to be copied; thus it is possible to move any size blocks of memory. The program could have used the Y register in a similar way, but the post-indexed mode can be faster and more flexible.

```
      LDY #0          ;Start with
                      ;zero.
LOOP  LDA (POINT1), Y ;Load byte.
      STA (POINT2),Y  ;Save byte.
      INY             ;Inc.index.
      BNE SKIP1       ;Skip if not
                      ;zero.
      INC POINT1+1    ;Else Inc.
                      ;high bytes.
      INC POINT2+ 1   ;of pointers.
SKIP1 /   /
      /   /           ;Perform
      /   /           ;some test.
      /   /
     "branch to"LOOP ;Go back if
                      ;more to
                      ;copy.
```

This program will work a lot faster because less memory accesses are being performed in the loop. This time we increment the Y register instead of the low byte of the pointer to copy upto 256 bytes, once the Y register goes back to zero we simply increment the two high bytes of the pointers to move along another 256 bytes into each table. As you might have guessed, this post-indexed mode is used far more often than its pre-indexed counterpart on the basis of saving speed and bytes in the program, a useful consequence is that the Y register may be re-loaded with a new index, used, and then the old index restored, without having to touch the pointer.

KEEPING TRACK

The second reason for using pointers, even when accessing 256 bytes or less, is that tables of data may not be in a fixed position in memory. Consider the problem of building several tables in memory, each of an undetermined size. You could start each table at a fixed address and then hope that one table doesn't overwrite the next, even if they don't you could well be wasting a lot of memory unnecessarily. The solution would be to set the pointer to the first table at the start of available memory, once built, the next free address would be stored as the pointer to the second table and so on, in this way the memory will be used more efficiently. A similar problem is when a subroutine has to be able to work on an arbitrary table, the calling program could simply save its pointer to its tables into the subroutines pointer, and then call the subroutine.

CRACKING THE CODE

continued from previous page

Pointers are used by the Atari operating system for one main reason; the amount of memory in the system could be different. For instance, the screen memory in a 16K machine starts at a different place than in a 48K machine, as a consequence a pointer is kept at the start of the screen; it then doesn't matter to the operating system where the screen is actually located, it will always access it correctly. It should be stressed that it is bad programming practice to assume that some area of memory is fixed when it could in fact be in a different place (I have seen programs that were written for a 48K machine and when run on a 16K machine, although there was enough memory, a blank screen appeared because the screen memory had moved!).

ACCESSING THE SCREEN

The reason why I have chosen to start with the screen is that it is an effective way of demonstrating the indexed-indirect accessing with instant visual results for you to relate to, and is a practical situation.

The short program I left you with last time (Listing 4), when run, showed all the possible 256 characters at the top of a graphics 0 screen. It did this by using the operating system pointer to the start of the screen, which is located at 58 hex, the Y register was used to index upto 256 locations into the screen, and at each location it stored the index value by transferring the Y into the accumulator.

THE SCREEN LAYOUT

Each of the standard graphic and text modes displays a fixed number of bytes per line and fixed number of lines down the screen. In common with all these modes is a simple mapping from a contiguous memory area onto the physical screen. The first byte of the memory area is mapped to the top left hand part of the screen, the next byte is mapped to the right of the last one, and so on, until one complete line is displayed, the consecutive bytes are then mapped to the start i.e. left side, of the next line, and so on, until the whole picture is displayed. Figure 3 shows this mapping for a 40 byte by 24 line display, such as graphics 0, with the byte offsets to each screen position, from the start of the screen memory, in bytes, shown inside the screen positions.

CHARACTER & BIT MAPPING

The Atari hardware allows for two methods of displaying screens; character and bit mapped, the difference being how the bytes from memory are displayed; both have the effect of turning a set of picture elements, or physical pixels, into a particular colour on the screen. A pixel is the smallest addressable unit of the display.

In a character mapped mode, such as graphics 0, each byte read from memory causes a matrix of physical pixels to be set, in this case it covers an 8 by 8 pixel area. The exact pattern of the character is determined by the character set being used.

A bit mapped mode splits the byte into a number of bit groups, or fields, each with the same number of bits i.e. eight by 1 bit four by 2 bits, two by 4 bits or one whole byte.

Each group of bits then determines the state of one logical pixel on the screen. Note that a logical pixel is a group of physical pixels, which can be treated as an entity; because only the hardware addresses the physical pixels we will abbreviate the logical pixels we address to just pixels or points. The distinction is not very important, just as long as you understand that a group of bits in a byte are controlling the state of one area, of some dimension, on the screen. For example graphics 8 has 40 bytes per line and 192 lines per screen, each byte is split into eight bits, each bit either turns a pixel on or off (one colour or another), giving 40x8, or 320 pixels per line by 192 down.

USING A BIT MAPPED SCREEN

Listing 1 accesses the bit mapped screen of graphics 8 via the operating system pointer at 58 hex, used for all modes to produce a random moving pattern.

CONSOL and RANDOM are labels set up to access special locations in the hardware, all you need know for now is that reading from CONSOL gives the status of the START, SELECT and OPTION keys and reading them from RANDOM produces a "random" 8 bit number. SAVMSC is the name given to the pointer to the start of the screen memory. PZERO and TEMP are variables used by the program, they have been assigned the values of $CB and $CD respectively. Straight forward equates could have been given e.g. PZERO = $CB, but instead I have set the location counter to $CB and defined the label PZERO. *=*+2 assigns the value of the location counter, *, plus two back to the location counter, thus moving it forwards by two, the same occurs for TEMP, giving it the value of $CD. This forward movement of the location counter is often referred to as "define space" and on some assemblers there is a special directive to perform this operation. The advantages of this method are that you can instantly see how many bytes are intended for each storage area, two in this case, and that you do not have to do any calculations for the value of each label, lastly the whole block can be moved to another area by simply changing the original value before the labels - a great time saver if there are a lot of variables!

Now onto the program. The code starts at 600hex, the first thing done is to remove the byte BASIC gave us on the stack. So as not to change the contents of SAVMSC, the two-byte pointer is copied into PZERO and PZERO+1. A random number is then loaded into the accumulator and tested against the number of lines down our screen - if it is greater than or equal to 192 then another number is picked, when the loop is finished A will contain a number between 0 and 191 inclusive. This Y co-ordinate is then saved in TEMP and the high byte of TEMP (TEMP+1) is zeroed, the code from lines 280 to 500 then multiplies this Y value by 40, the number of bytes per line, to give the offset to the start of the given line, which is then added to our pointer to the start of the screen, PZERO.

For speed and convenience the method by which TEMP is multiplied by 40 is to multiply it by 8, add it onto PZERO, and then to multiply TEMP by a further 4 to give a total of 32 times which is also added to PZERO. The 8 times TEMP plus the 32 times TEMP gives the 40 times TEMP we needed to add PZERO. The multiplications are all performed on TEMP and

CRACKING THE CODE

continued from previous page

TEMP+1 using shift and rotate operations, multiplying by two each time.

Another random number is now loaded but this time it has to be between 0 and 39, so we compare it to 40 and get another number if the carry is set. To speed the selection process, the top two bits of the ramdom number are zeroed by AND with a mask of 00111111 or $3F, giving a number between 0 and 63, thus more of a chance it will be in the right range (15% verses 63% in theory!). This value is used as the byte index into the selected line and is moved to the Y register. Yet another random number is loaded and the 8 bit pattern is stored into the selected byte of the screen, pointed at by PZERO and offset by the Y register. By changing the byte, 8 pixels will set to on or off in that area. Lastly the program loads the value of CONSOL and checks it against 7 (all keys up), if it is equal the program starts again, otherwise it stops and returns to BASIC.

To run the program either type in the BASIC program. Listing 2, or assemble the source code of Listing 1 and load the object code. Once the program is loaded from 1536 onwards, it can be run by typing;
GR.24:X=USR(1536) from BASIC. If the screen fills up with random rubbish, dont worry; that means the program is working! To abort it press either the START, SELECT or OPTION key.

BASIC LOADERS

To finish this time we have five BASIC programs, the one last time read a binary file from disk or tape into memory, these five are short utilities to create data and to read data for use in a BASIC program.

Listing 3 asks for a device to write its file to, again, for cassette use C:, and for disk D: followed by a filename, it will then ask you for four more decimal numbers. Assuming you have got assembled code sitting in memory, the program will create a file containing line numbers and DATA statements. The data in the area you specified by the start and end addresses will be placed after the DATA statements and the line numbers will start at the number you gave and increment by the number you gave. The program will then tell you how many bytes had been read from memory. The file, once created, can then be ENTERed over any existing BASIC program which can then read in the data, such as Listing 4, note that if you know how many bytes are to be read it is easier to use a FOR/NEXT loop.

Listings 5 and 6 are similar to those in 3 and 4. Listing 5 makes a file of DATA statements but lists its data in strings of hex characters, which can then be read by a program such as that in Listing 6. If you haven't got an assembler then listng 6 is particularly useful as you will be able to write hex characters on the DATA statement lines in the following way:

```
10000 DATA A460FF458296D51ABC579830
```

The twenty-four characters represent the twelve hex numbers to be placed in consecutive locations, you could write more or less characters per line, but this is the number of data which Listing 5 generates. Lastly, Listing 7 creates a file containing a string dimension and then some string assignments.

If any quote or carriage return characters are found, then separate assignments are made for them using the CHR$ function. If you use this method you must make sure that the program which is placed in the string, MC$, is relocatable e.g. there are no jump, or jump to subroutine, instructions as the program will crash. To call the code in a string use the string address function:

```
X=USR(ADR(MC$))
```

Hopefully I have covered enough short utilities in BASIC for most of your purposes and it shouldn't be to difficult to modify the routines to suit your own needs.

Next time we will start a detailed examination of the hardware and operating system facilities.

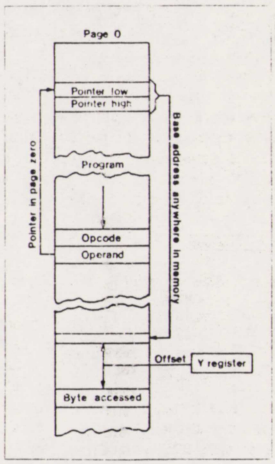


Figure 2. Post-indexed e.g. LDA ($CB),Y

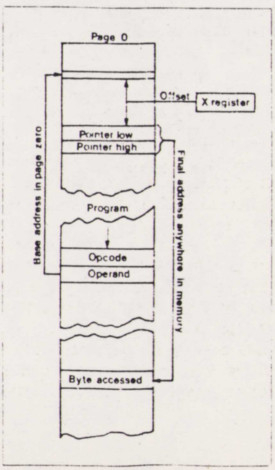


Figure 1. Pre-indexed e.g. LDA ($CB,X)

CRACKING THE CODE

continued from previous page

```
0100 CONSOL  =     $D01F   ;Read START/SELECT/OPTION.
0110 RANDOM  =     $D20A   ;Random number generator.
0120 SAVMSC  =     $58     ;Screen pointer.
0130         *=    $CB     ;Start of variables.
0140 PZERO   *=    *+2     ;Pointer to screen.
0150 TEMP    *=    *+2     ;Temporary locations.
0160         *=    $0600   ;Start of program.
0170         PLA           ;Clean up stack.
0180 START   LDA   SAVMSC  ;Copy low byte
0190         STA   PZERO   ; of pointer.
0200         LDA   SAVMSC+1;Same with high.
0210         STA   PZERO+1
0220 NEWY    LDA   RANDOM  ;Get random number.
0230         CMP   #192    ;If 192 or more
0240         BCS   NEWY    ; get another one.
0250         STA   TEMP    ;Save as low byte of Y.
0260         LDA   #0
0270         STA   TEMP+1  ;Zero high byte.
0280         LDX   #3      ;Count of 3.
0290 MULT8   ASL   TEMP    ;Multiply by 2...
0300         ROL   TEMP+1
0310         DEX           ;Dec. count and branch.
0320         BNE   MULT8   ; to give 8 times TEMP.
0330         LDA   PZERO   ;Add low byte to PZERO.
0340         CLC
0350         ADC   TEMP
0360         STA   PZERO
0370         LDA   PZERO+1 ;Add high byte.
0380         ADC   TEMP+1
0390         STA   PZERO+1
0400         ASL   TEMP    ;Multiply TEMP
0410         ROL   TEMP+1  ; by 4
0420         ASL   TEMP    ; to give 32 times
0430         ROL   TEMP+1  ; as total.
0440         LDA   PZERO   ;Add to PZERO again.
0450         CLC
0460         ADC   TEMP
0470         STA   PZERO
0480         LDA   PZERO+1 ;Add high byte
0490         ADC   TEMP+1  ; to give 40*Y value.
0500         STA   PZERO+1
0510 NEWX    LDA   RANDOM  ;Another random number.
0520         AND   #$3F    ;Zero top 2 bits.
0530         CMP   #40     ;If 40 or more
0540         BCS   NEWX    ; get another.
0550         TAY           ;Save as index into line.
0560         LDA   RANDOM  ;Another random number.
0570         STA   (PZERO),Y;Save on screen.
0580         LDA   CONSOL  ;Test buttons.
0590         CMP   #7      ;All up?
0600         BEQ   START   ;Yes - go again.
0610         RTS           ;Else return to BASIC.
```

Listing 1

```
QZ 10 DIM HEX$(16)
CV 20 LINE=10000:TRAP 100:J=0:START=1536
VA 30 READ HEX$,CHKSUM:SUM=0
AA 40 FOR I=1 TO 15 STEP 2
JG 50 D1=ASC(HEX$(I,I))-48:D2=ASC(HEX$(I+
   1,I+1))-48
KT 60 NUM=((D1-7*(D1)16))*16+(D2-7*(D2)16
   )))
LW 70 SUM=SUM+NUM:POKE START+J,NUM:J=J+1:
   NEXT I
LY 80 IF SUM=CHKSUM THEN LINE=LINE+10:GOT
   O 30
IN 90 ? "Checksum error on this line:"
VO 95 LIST LINE:END
YS 100 PRINT "Data in memory."
TB 10000 DATA 68A55885CBA55985,1000
KF 10010 DATA CCAD0AD2C9C0B0F9,1415
LK 10020 DATA 85CDA90085CEA203,1011
XV 10030 DATA 06CD26CECAD0F9A5,1279
WF 10040 DATA CB1865CD85CBA5CC,1238
TG 10050 DATA 65CE85CC06CD26CE,1099
RB 10060 DATA 06CD26CEA5CB1865,948
MN 10070 DATA CD85CBA5CC65CE85,1350
PO 10080 DATA CCAD0AD2293FC928,942
DI 10090 DATA B0F7A0AD0AD291CB,1332
JG 10100 DATA AD1FD0C907F0AA60,1126
```

Listing 2.

```
XQ 10 DIM FILE$(14)
FW 20 ? "Please enter device/file name"
CH 30 ? "for output..."
AR 40 INPUT FILE$
LY 50 OPEN #1,8,0,FILE$
GK 60 ? "Please enter start address..."
ZM 70 INPUT ST
YS 80 ? "Please enter end address..."
RQ 90 INPUT EN
GB 100 ? "Please enter starting line..."
IC 110 INPUT LIN
WM 120 ? "Please enter line increment..."
DU 130 INPUT INC
ZB 140 CH=0
MH 150 FOR I=ST TO EN
DV 160 IF CH=0 THEN PRINT #1;LIN;" DATA "
   ;
DP 170 IF CH=7 OR I=EN THEN PRINT #1;PEEK
   (I):GOTO 190
RJ 180 PRINT #1;PEEK(I);",";
VC 190 CH=CH+1:IF CH=8 THEN CH=0:LIN=LIN+
   INC
FS 200 NEXT I
LA 210 CLOSE #1
GF 220 ? "Number of data = ";EN-ST+1
```

Listing 3.

```
DU 10 TRAP 40:LOC=1536
QH 20 READ NUM:POKE LOC,NUM
DF 30 LOC=LOC+1:GOTO 20
YS 40 END
```

Listing 4.

```
NZ 10 DIM FILE$(14),HEX$(24),CONV$(16)
HL 20 CONV$="0123456789ABCDEF"
FX 30 ? "Please enter device/file name"
CI 40 ? "for output..."
AS 50 INPUT FILE$
LZ 60 OPEN #1,8,0,FILE$
BL 70 ? "Please enter start address..."
ZN 80 INPUT ST
YT 90 ? "Please enter end address..."
UJ 100 INPUT EN
GD 110 ? "Please enter starting line..."
IE 120 INPUT LIN
WO 130 ? "Please enter line increment..."
DW 140 INPUT INC
JI 150 LN=EN-ST+1:LOC=ST
GO 160 NUM=12:HEX$=""
GE 170 IF NUM>LN THEN NUM=LN
MA 180 FOR I=1 TO 2*NUM STEP 2
IM 190 X=PEEK(LOC)
YY 200 D1=INT(X/16):D2=X-D1*16
PH 210 HEX$(I,I)=CONV$(D1+1,D1+1):HEX$(I+
   1,I+1)=CONV$(D2+1,D2+1)
HH 220 LOC=LOC+1
FY 230 NEXT I
WP 240 PRINT #1;LIN;" DATA ";HEX$
RC 250 LIN=LIN+INC:LN=LN-12
MG 260 IF LOC<=EN THEN 160
GP 270 ? "Number of data = ";EN-ST+1
LO 280 CLOSE #1
```

Listing 5.

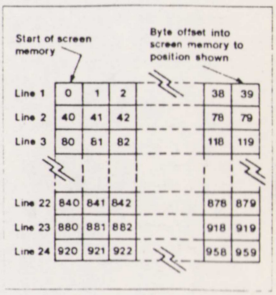


Figure 3. Screen Layout

CRACKING THE CODE

continued from previous page

```
QK 10 DIM HEX$(24)
DE 20 TRAP 100:LOC=1536
BS 30 READ HEX$
VK 40 FOR I=1 TO LEN(HEX$) STEP 2
ZG 50 D1=ASC(HEX$(I,I))-48:D2=ASC(HEX$(I+
   1,I+1))-48
KT 60 NUM=((D1-7*(D1>16))*16+(D2-7*(D2>16
   )))
VO 70 POKE LOC,NUM:LOC=LOC+1
IW 80 NEXT I
SI 90 GOTO 30
WQ 100 END
```

Listing 6.

```
JK 10 DIM FILE$(14),CRS(50),QTS(50)
FW 20 ? "Please enter device/file name"
CH 30 ? "for output..."
AR 40 INPUT FILE$
LY 50 OPEN #1,8,0,FILE$
GK 60 ? "Please enter start address..."
ZM 70 INPUT ST
YS 80 ? "Please enter end address..."
RQ 90 INPUT EN
GB 100 ? "Please enter starting line..."
IC 110 INPUT LIN
WM 120 ? "Please enter line increment..."
DU 130 INPUT INC
SS 140 LN=EN-ST+1
JH 150 PRINT #1;LIN;" DIM MC$(";LN;")"
WR 160 PSS=1:PSE=50:FIN=0
LV 170 CR=0:QT=0:LIN=LIN+INC
FJ 180 IF PSE>LN THEN PSE=LN:FIN=1
GZ 190 PRINT #1;LIN;" MC$(";PSS;",";PSE;"
   )=";:PUT #1,34
AL 200 FOR I=PSS TO PSE
DH 210 X=PEEK(ST+I-1)
UN 220 IF X=34 THEN QT=QT+1:QTS(QT)=I:X=3
   2
CK 230 IF X=155 THEN CR=CR+1:CRS(CR)=I:X=
   32
SB 240 PUT #1,X
GC 250 NEXT I
BZ 260 PUT #1,34:PUT #1,155
GS 270 IF QT THEN GOSUB 320
GF 280 IF CR THEN GOSUB 370
VU 290 IF  NOT FIN THEN PSS=PSS+50:PSE=PS
   E+50:GOTO 170
KZ 300 CLOSE #1
NU 310 END
ZA 320 X=34
HZ 330 FOR I=1 TO QT
ZA 340 P=QTS(I):GOSUB 420
GD 350 NEXT I
ZL 360 RETURN
RN 370 X=155
YL 380 FOR I=1 TO CR
VA 390 P=CRS(I):GOSUB 420
FU 400 NEXT I
ZC 410 RETURN
ZY 420 LIN=LIN+INC
FD 430 PRINT #1;LIN;" MC$(";P;",";P;")=CH
   R$(";X;")"
ZI 440 RETURN
```

Listing 7.

DISK CONTENT

Side A of this month disk contains a card game, called HEARTS.OBJ, to load this game you must disable BASIC. And use the 'L' Option from DOS, or copy the program to another disk and rename it to AUTORUN.SYS.
You play this card game against four computer players, if you like to play cards this is for you. The full documentation is on the disk as HEARTS.DOC.

PUZZLE PLEASURE is a board type game, it consists of 16 squares with the alphabet from 'A to O'. One square is empty to let you move the characters, the aim is to set up all the characters in alphabetical order. The movement of the characters is done by just typing the number in which the character is situated. The prompt on the screen is asking 'Move from' and after pressing return the prompt will ask 'Move to', it will also count the moves it takes you to complete this puzzle. This is also a binary load, the 'L' option can be used to run this program.

TRICKY.OBJ is rather a tricky game, the aim is to keep the BUG from hitting the side of the screen or falling of the track you create by moving one square with the joystick plugged into port one. Again use the 'L' option to load and run, you can press the joystick button or the Start key to view a demo of the Skuttlebug moving around the screen. When you've seen enough just press the Space Bar for some instructions.

There is also a digitized sound picture program on which must be loaded with BASIC disabled then usind the 'L' option. When loaded you press the Start key for the digitized speach.

There is a DEMO by the Author of the new book telling you about it, read all the text. When pressing the appropriate key to run the demo on side A you will be asked to flip the disk.

A BASIC program called TEXTVIEW.BAS when run will read all the DOC. files on side B. It contains utilities and all have doc files. You can also use the file reader on side B to read the instructions.

TWAUG PUBLICATIONS PRESENTS

THE Atari XL/XE
Complete And Essential MAP Including Probably The Most Comprehensive Appendice Selection Ever Produced

Written by Andrew C.Thompson

This Book Contains Information Never Released Anywhere Before And Is Heavily Based And Expanded On Mapping The Atari - Revised
If you haven't read the book extracts article in issue 65 of New Atari User, then we would be very pleased to send you a copy. Just send us a cheque or postal order for £1 to cover production and postage costs.
We hope to release the book by the end of February. We would like to thank all those who have already written to us asking for their names to be put down for a copy when released. We will inform you when the book is ready, by letter and through New Atari User.

DAISY-DOT III + PAGE 6 WRITER + SPARTADOS

equals Max's emulation!

I use DAISY-DOT III quite a bit and ever since I read Max's excellent series on DAISY-DOT III and TEXTPRO and the way he is able to put everything into RAM, I've wanted to be able to do the same. So what's the problem?

The problem is, that Max has a 1 meg upgrade and I have a standard 130XE and a single, standard 1050, and with this Government (Petticoat) and in the current financial climate (retirement) there is not a snowball's chance in Hades of my equipment equalling his.

With my limitation on memory it was obvious that I could not load DD3, TEXTPRO and a host of fonts into 64K RAM and still have room to put text in there as well. The following is an account of my efforts to try to emulate Max using my standard rig.

The only way I could save on memory was to use a different, more compact, word-processor - but which? I had recently put SPEEDSCRIPT on to tape for the daughter of a friend who wished to practice her typing and I had used it to write a foreword to the instructions regarding cassette use. I was therefore aware that it was compact, (only 67 sectors long), surprisingly powerful for its size and not dissimilar to TEXTPRO in its basic functions, so I loaded it up and quickly discovered two problems. Pressing (SHIFT +) would not enable the backslash which is essential for DD3 and there was no facility to recall DOS. I also knew that PAGE 6 WRITER was very similar so I loaded that. (Could it be that SPEEDSCRIPT is the parent of both TEXTPRO and P6 WRITER?). Again no backslash but, although not documented, CONTROL Q gave the choice of returning to DOS.

I next tried my favourite, 1st Xlent WP, copying only the WORDPRO file, without HELP screens etc. This does support the backslash but without its ancillary files it would not recall DOS and at almost twice the length of SPEEDSCRIPT and P6 WRITER was running the memory banks rather short.

I left the problem for a while then thought I might see if the PP.MAC file from DD3 would work with P6 WRITER, but when I reloaded P6W I found (by accident) that if I first pressed ESC before the (SHIFT +) combination that the backslash was enabled. This is a small price to pay for the ability to put the whole lot into RAM, and even if the result is much more modest than that described by Max, it is very usable.
The way I did it was as follows:
I formatted a disk (single density) using SPARTADOS X32D and copied to it RD.COM (to create a RAMdisk), PP.COM from DAISY DOT III, PAGE 6 WRITER (the AUTORUN.SYS file from the boot disk), changing its name to WP.COM, and three fonts. A STARTUP.BAT file was created from the SPARTADOS 'D1:' screen by typing:
COPY E: D1:STARTUP.BAT (RETURN)
This gives a blank screen with the cursor at the top left position. The batch file reads:

```
RD D8:
COPY D1:PP.COM D8:
COPY D1:WP.COM D8:
COPY D1:*.NLQ D8:
D8:WP
```

After typing the above, a (CONTROL 3) writes it to the disk - ensure the correct disk is in D1 at the time.

Perhaps a more obvious way of producing the STARTUP.BAT would be from the word processor and saving it to the disk. This is also an easy way to edit the file should it ever be necessary.

When the disk is booted (with OPTION), the RAMdisk is created and the PP and WP files are loaded into RAM along with any fonts with the .NLQ extender then the default drive is changed to D8 and WP.COM is loaded from there (almost instantly). At this stage, with the three fonts I used, there are 143 free sectors in the RAMdisk.

After typing in your text, SAVE it to D8 by pressing CONTROL S and entering 'D8:filename.ext'

If CONTROL M is pressed, a DIRectory of D1 is shown. If then 8 is pressed a DIRectory of D8 is shown and this becomes the default whenever CONTROL M is used.
Pressing CONTROL Q brings 'QUIT TO DOS Y/N' on the command line. A 'Y' will bring up the 'D8:'screen. At this point if PP(RETURN) is entered (No need for the .COM extender), then DD3 will load instantly and can be used to print out your masterpiece.

You may return to the word processer at any time by pressing CONTROL X to exit DD3 for DOS and typing WP(RETURN).

Don't forget when typing text which is to be printed with DD3, to prefix font changes with 'D8:', The same applies when naming the starting font and the file to be printed at the start screen of DD3.

Using SPARTADOS with DD3 has a further bonus; it can be set up for automated printing and works as follows:

If you wish to print a file called 'TEXT' using the font 'POET', then instead of simply typing 'PP' to load DD3, type 'PP POET TEXT' and, providing all three items are available on the default drive, printing will take place automatically and you will be returned to DOS. This system can also be set up as a BATCH file which must be typed as shown (horizontal line of print) and called, perhaps 'TEXT.BAT'. The transfer of this batch file to D8 can be arranged on the STARTUP.BAT and when it is required all that is necessary is to type '-TEXT' (don't forget the dash) when the D8: screen is on. This procedure can be very useful if you have to make regular copies of the same document.

If there are any other duffers like me out there then I hope this article will be of benefit and save you the hassle of finding out the hard way what so many others find so easy.

Oh! and by the way, don't forget to save your text to a floppy in your eagerness to try this system - as if anyone would...!

by John Bunting

TEXTPRO WORD PROCESSOR

by Max Gerum

I'll begin part 3 of TextPro by continuing with the Editor commands. Firstly let me explain how to change the editors background colour, I've had a number of enquiries about this function. When using TextPro 4.54 press CTRL+B and you move through the colours to reverse the sequence press SELECT+CTRL-B. To change the brightness you press CTRL-T and to reverse the sequence press SELECT+CTRL-T. When using TextPro 4.56 the combination of keypresses mentioned above selects the BANKS, to change the screen colour use CTRL-> or CTRL-DELETE key and SELECT+CTRL-> or DELETE key. The colour can also be changed in the CNF file, but I would only recommend this method when you know what you're doing. After you've chosen the colour you want for the editor, press SELECT+CTRL-S(ave config)D: type TEXTPRO.CNF, press Return and the chosen colour will boot with TP.

Saving the configuration file seem to cause some difficulties by the enquiries I received. When you save the config file with the SELECT+CTRL-S combination you save everything you've changed. For instance, we'll say you use a RAMdisk as D8: and the last key you pressed was CTRL-M and then 8 before you saved the configuration, what will happen is that when you re-boot TextPro and you press CTRL-M to have a look at the directory, the RAMdisk directory will be displayed first. The following changes are also saved, for instance, if the editor is in either Insert or Replace mode you toggle between these two modes using CTRL-I. If the characters are in Upper or Lower case, use the Caps key to change from one to the other. The fast Repeat cursor key, use the CTRL-N combination to toggle on or off, and SELECT+CTRL-N selects the key Click on/off.

There is another selection menu you can call to cycle through some option. Pressing SELECT+CTRL-;... semi-colon - selects Verify on/off option, pressing the Return leaves this option off. After you've pressed return the next prompt is Menu sorting on or off, press either 'Y' or 'N', pressing 'N' will sort the directory alphabetically on screen and this way is much easier to read. The next option is SIO sound on/off, answer 'Y' to turn off the SIO bleeping. The next prompt is New EOL, when you press 'Y' you must enter your new character, but if you only use your TextPro for Atari material the default return character should do. Now if you have entered all these changes and save it to the config file with SELECT+CTRL-S(ave) config to the CNF file, you will find when booting TP, everything you've saved into the config file is now in memory.

There is one feature I do like in TextPro, I mentioned this option in issue #4, and that is the CTRL-W(here's) the cursor. Setting up a text file to print out with DD3 is quite easy to do by using the CTRL-W. Daisy Dot 3 prints text a different way than TextPro does, the page lines do not correspond with each other. By printing the first page with DD3, setting up the ODD page printing function, then returning to TextPro and checking how many lines DD3 printed, by using the CTRL-W combination, this will tell you the exact number of lines for setting up other pages of text.

The default page setup for TextPro is 58 printing lines, now you can change these lines to suit the length of the paper you are using. Most of you probably leave a space of one inch at the top and the same amount at the bottom, so you would set the lines for printing to 54 lines. Now every time you use the CTRL+W key press, the display in the command line at the top of the screen would read "Page 1, Line 10 of 54", with page length of 66, this is only an example. But, as I mentioned above you can change the length of the lines by setting the bottom margin using the inverse "b". Holding the SELECT key and typing <b> will print the lower case "b" in inverse, now without a space type the printing lines you want, you would naturally type the setting in the first line before commencing to type in your text.

TEXTPRO WORD PROCESSOR continued

I have mentioned the CTRL-W feature again just to let you know that it can be used when writing text to be printed with Daisy Dot. DD3 prints more lines per page than TextPro does, it does with me, the reason is I close the lines up a little to get more words onto a page.

TextPro also allows you to edit graphic characters directly into a text file using the "ESCape Set" mode. One of the most handy applications is cleaning up ASCII files. ASCII uses a decimal 13 or CONTROL-M for a Carriage Return (CR) and decimal 10 or CONTROL-J for a Line Feed (LF). When you view an ASCII file in TextPro, these CR's and LF's will appear as an outline character M and J. To print these characters on your screen first press ESCape than press CTRL-M, again ESCape and CTRL-J. On the other hand, Atari 8-bit computers speak ATASCII, which uses decimal 155 as a combination LF and CR.

It's easy to convert these ASCII characters into ATASCII using TextPro's Global Search and Replace command, CONTROL-G, while in the editor with an ASCII file. This feature has also been mentioned in part two of my article TextPro. I am certain some of you have encountered these ASCII CR's and wished you could change them to the ATASCII CR quickly and didn't know how. Let me explain it in detail, so anybody who is learning to use this marvellous word processor, knows exactly what to do.

Move the cursor to the beginning of the file and press CONTROL-G. The command line now displays the prompt "Find:" Now hit the ESCape key, (ESC) and type CONTROL-M. Hit (ESC) again and type CONTROL-J. Notice that the funny characters in the status line look just like the ones in the ASCII text file. Now hit RETURN, "Change to:" will now appear in the command line. Hit ESC again and type CTRL-+, you will now notice this symbol looks exactly the same as the Return character used by TextPro, "a square left arrow". Now hit RETURN, TextPro will now replace all those funny looking characters with civilized CR/LF'S. I use the Find and Replace option quite often, as I do receive a lot of articles with the ASCII characters in.

TextPro uses macro files and I will explain next in more details and give a couple of examples how to create them. TextPro version 4.54 and earlier use macro files with the extender .MAC, for macro files. Because of some changes in the use of some macro command keys, the extension .MAX is used in TPX 4.55 and later. TextPro version 4.56 macro files use the extender .MAX, this version is in the TWAUG PD library.

What are Macro's?

Macros are simply designated keys that will type many keys for each macro key you type. If you are using TextPro you will find some macro files on the TextPro disk. I am saying 'some', you will have TEXTPRO.MAX or if you have the version 4.54 or older the extender .MAC and maybe a file called PP.MAC, or a number of others.

The help files are an example of a simple macro. You press OPTION and a number and the macro types the keys necessary to copy a particular file from the disk to the editor. As an example: You press OPTION-1 and the macro types:

SELECT+CONTROL-Q (Q is the copy command)
TPHELP.01,E: (filename, E: is the Editor)
RETURN (execute the copy command)

In this case, the macro typed 14 keys while you only typed 2.

First, you have to load a macro into the macro buffer. Macros are contained in disk files, which are text files created by typing the correct macro information in the TextPro editor and saving it to disk.

TEXTPRO WORD PROCESSOR continued

Macro files use the extender .MAX, or .MAC for older versions, to identify them as such. There are three ways to load a macro file into the macro buffer.

1- Rename the macro file TEXTPRO.MAX, or .MAC and it will load automatically when you load TextPro from disk.

2- From the TextPro editor, use CONTROL-V and type the full macro name, then type Return.

3- Macros can be loaded by pressing START and typing the macro name, without adding .MAX. With TEXTPRO.MAX in the macro buffer, START calls a macro to load another macro. Now this paragraph only refers to version 4.55 on. I give you a good example: TEXTPRO.MAX must be in the macro buffer, you then hit START and in the command line the prompt "MACRO D:" is printed type XLOAD, which is a macro file on the TextPro version 4.56, "DO NOT ADD .MAX". When you press Return the macro exits the input mode and adds the extender .MAX to the filename. Again, I am using a function for examples which I wrote about in issue #5, but here I am explaining the macro side of it.

Once this XLOAD.MAX macro file is loaded it will ask for the filename of the file you want to load into the editor. You use this macro file when you have a very large document that needs segmenting. Instead of using the 'C' Parameter command to segment a file, this macro file will segment a file into seperate Banks.

Let me give you another example on how to create a macro, for an example, to Indent Paragraphs:
Lets define the RETURN key as the macro key

<RETURN> followed by <SELECT => defines the macro key. We want the feature to add an additional return character and a five space TAB for indent of the paragraph so we enter:

Hit <RETURN> and <RETURN> then <ESC><TAB>
If this sequence is included as part of a macro file and you press <OPTION+RETURN> the program will insert the two carriage returns followed by the TAB. We could also add a prompt to the macro that would print "New Paragraph" into the command line. Just press:
<SELECT+CTRL-Y>New Paragraph<RETURN>

Now every time the macro <OPTION+RETURN> is executed the prompt "New Paragraph" will appear on the command line.

If you have the macro file DD3.MAC on your TextPro disk this macro is included in that file. Yes, DD3.MAC is a large macro file, it includes all the codes used in Daisy Dot 3.

Again I like to show you a couple of examples. For instance when you set up an icon for printing you use characters thus, !"#, $%& and so on. Well by pressing OPTION-1 to OPTION-0 the first ten character set prints out onto the screen. When you want to include a new font into a text you would enter a backslash \ and the F for font, when DD3.MAC is in the buffer all you have to press is OPTION-F. I use more than one drive and the fonts I use are resident in D8:, so I changed the F to print \FD8, when I press OPTION-F and all I do is type the colon to the end and of course the fonts filename.

With me using the macros most of the time, I've set it up to load in the macro buffer automatically when TextPro is loaded. And this is how I've done it, for those readers who don't know how it's done:

TEXTPRO WORD PROCESSOR continued

Press SHIFT-8 = @ next hold SELECT and press the equal sign <=>, this sign is now in inverse now press ESCape <ESC>CONTROL-V. Before you can print the next sign to represent the device you must hold CTRL and press the Caps key, this switches the commands off, done that press DELETE BACK SPACE. Now press CTRL-CAPS again to switch the commands back on again and type the number, for the drive that's holding DD3.MAC, then type DD3.MAC.

Now every time TextPro is booted the macro file loads into the macro buffer. Now the best place to type this macro to is directly into TEXTPRO.MAC. Load TEXTPRO.MAC into the editor, you are now able to edit the MAC file at your leisure and you can place the macro explained above anywhere you like.

Enjoy TextPro.

TWAUG AT AMS7.

I am pleased to say that TWAUG's first visit as a user group to the AMS show at Bingley Hall back in November was a great success. We look forward to many more.

It was great meeting up with many old friends, and also making many new friends. It was a shame that Max was unable to attend the show as he had just came out of hospital after his big operation, but we must give our thanks to Tommy Wood who stood in for Max and gave us his support by helping John and myself on the stand. Thanks Tommy from all at TWAUG.

We would also like to give our thanks to John Bunting for giving us copies of the photographs that he took at the show. Above are two photos, the one of the HANDSOME chap standing on his own is yours truly, the other is John and Tommy.

John is the one looking into the disk box, Tommy is standing beside him (still with his coat on), I think he was feeling the cold.

We have received a letter from Sharwood Services telling us that they are going to hold another show in April which will be called SAMS (Spring All Micro Show). There will still be the usual AMS show in November and both will be held at Bingley Hall Stafford.

Well that's all from me, I'll let you get on reading the rest of the newsletter. All the best to you all for 1994 from all at TWAUG.

David Ewens.

PICMAN

by Ed Hall

(FR)ANTIC, Sept. 1993
Alamo Area Atari User Assoc.

One of the most fascinating aspects of our Atari computers is their graphics capability. Again and again I've been astonished by what they can do--just look at some of the incredible demos coming out of Europe in recent years.

For those interested in creating their own images, there is a wonderful assortment of art programs available--EscalPaint, Graphics Art Department, and Technicolor Dream are just three. These and others can use hundreds, thousands, and even millions of colours. (Yes, you read that correctly. Technicolor Dream claims a palette of millions of colours).

But keep this in mind: an art program is nothing more than a tool, a glorified paint brush. It is up to you to supply the creative spark. Having millions of colours at your disposal does not guarantee a great picture; conversely, if you have the talent, you can produce interesting and attractive pictures even with clunky old programs like Micro-Painter and Micro-Illustrator (both of which use only 4 colours). Just remember this one rule: every art program has its own strengths and weaknesses. Learn what they are, and use them to their best advantage. For example, Technicolour Dream is particularly good for landscapes. As for Micro-Painter and Micro-Illustrator, avoid circles and work with rectangles as much as possible.

Something else to remember: because our computer has been around for a long time, there is a huge amount of public domain software available. Sure, some of it is bad, but some of it is pretty good. I've been collecting graphics utilities for years, and have put some of the more interesting ones on a disk called PicMan. Originally the title was short for "Picture Manipulator," but after a while "Picture Maniac" seemed a lot more appropriate.

Video Stretch

This program appeared in the October 1986 issue of ANTIC. Once the program is loaded, you can use paddles to stretch and distort a picture in real time. One paddle controls the size of the picture, the other selects the portion of the image to be seen. You can also use a touch tablet, but it doesn't work quite as well.

Press D to try out the built-in (not very interesting) demo picture in Graphics 9. Press L to load your own picture. Note that the program does not allow you to enter a filename; it will only load a picture named PICTURE. Because of this limitation, the PicMan disk has only two pictures (one on each side) for use with Video Stretch.

Video Stretch is fun for a while, but since there's not much you can do with it, the fun is rather short-lived.

Picture Bender

In the August 1985 issue of ANTIC there appeared a similar program called Display Master. It allows you to load a picture, then do all sorts of things to it: invert, flip, roll, fold, and other neat stuff.

Picture Bender is Display Master with a few minor changes. The commands are:

A. RESTORE. Returns picture to its original form.
B. INVERT. Flips picture vertically.
C. MIRROR. Flips picture horizonally.
D. SHIFT. Moves picture sideways the distance of one half-screen.
E. DOUBLE. Reduces picture to half its size vertically, then produces two copies and positions them one above the other.
F. FOLD. Doubles the size of a picture vertically, then folds it in half.
G. UNFOLD. Returns a folded picture to normal or, when used on an unaltered picture, doubles it and inverts one of the images.
H. ROLL. Flips picture vertically using a rolling motion similar to the Fold command.
I. WAIT. Pauses display for 2 seconds.
J.K.L.M. SCROLL COLOURS. Cycles the four colour registers.
N. SCROLL. Halves and quarters the picture horizontally, then scrolls each part in opposite directions.
O. REPEAT. Go to first instruction.

You can change instructions after they've been entered by using the following keys:

- Move back one instruction.
+ Skip to next instruction.
* Delete instruction.

To begin the show, press the = key.

Here are a few tips:

a) Don't forget the "D:" when entering a filename.
b) Use the "I" (wait) command often, so that viewers have a chance to take in the altered display.
c) Pictures may get really botched up if you alter them drastically (e.g. if you double them repeatedly), so until you get the hang of it, use the "A" instruction occasionally to return the picture to normal.
d) Though the program requires a lot of keypresses to set up, shows are often brief.

PICMAN continued

Therefore, always end your instructions with the "O" command, thus creating an infinite loop and giving you sufficient time to inspect your work. The "O" command is also useful when giving prepared demonstrations at club meetings and computer shows.

e) Hold down the START key to interrupt a show and return to the menu screen.

Pictures

Video Stretch and Picture Bender both require uncompressed 62-sector pictures. Since these take up a lot of space, the PicMan disk is in 1050 "dual" density in order to provide a good selection.

All the pictures are great, but some work better than others when squashed, rolled, or otherwise mutilated. To get you started, try the following Picture Bender commands with these pictures:

TITLE.PIC or GIANT.PIC - FGICFGIBIO
ALF.PIC or TPASTE.PIC - GICIFIBIO

You may also want to construct some pictures especially for use with Picture Bender. Try something which is eccentric (i.e. uncentered).

The Menu Screen

When I began putting together this disk, I wanted to create a custom menu screen which reflected the theme of the disk. I began by drawing two very simple pictures; one said PIC in huge block letters, and the other said MAN. Then I used a program called Hardwired Ray Tracing from the February 1989 issue of ANTIC. It joined the two pictures together and angled the letters of both words to slant towards a vanishing point in the middle of the screen. (You can see this picture on side 2. It's called HARDWIRE.PIC.)

After cleaning up the resulting picture, I went to work on the faces. That rather ominous-looking guy is actually Larry of the Three Stooges. Using Animation Station's cut-and-paste feature, I lifted Larry's face from the picture STOOGES.PIC (also on the disk) and stretched the poor guy's mug out of shape. Animation Station (aka Blazing Paddles) also has a "scroll" feature, which is very useful for shifting a picture about on the screen.

I did a lot of experimenting before I came up with a picture which I was satisfied with. In doing so, I utilized another handy program, this time from the August 1979 issue of ANALOG. Called Picture Perfect, it allows you to flip, halve, double (and even animate) a picture.

I've placed the Picture Perfect and Hardwired Ray Tracing progams on side 2 of the PicMan disk. However, neither is very friendly, so before using them you should refer to the magazine articles.

Also, note that Hardwired Ray Tracing takes quite a while to produce a picture, so make sure you use TurboBasic. (Even then you'll have time for a couple cups of coffee.)

By the way, did you notice how the menu picture pops up? This neat routine by John Collins appeared in the June 1993 issue of (Fr)Antic.

A Challenge

As much fun as Video Stretch and Picture Bender can be, they both suffer from the same obvious shortcoming--the inability to save a manipulated picture. One of the reasons for preparing this disk and writing this article is the fond hope that it might stimulate someone to take on the task of altering the original programs. You'll need an ACTION cartridge, since both programs were originally written in that language. The source code for both programs may be found on side 2 as STRETCH.ACT and DMASTER.ACT. I've also included the original runtime version of Display Master on side 2, as well as technical info on the workings of both ACTION programs.

Of course, there are lots of other features one could add. For Video Stretch, it would be nice to be able to stretch a picture horizontally as well as vertically; it would also be nice to enter a picture's filename for loading. But I'd be happy with any runtime version of either program which has a save capability. If anyone out there succeeds in doing the job, please let me know.

Ed Hall
66 Morrison Drive
Yellowknife, N.W.T.
Canada X1A 2Z1

Edwin Hall - Fidonet 1:3402/30
(The Really Board)

Notice: PICMAN a double sided disk, it is now in our Public Domain Library.

Happy Birthday TWAUG

HOORA!!! We have reach our first successful year with our newsletter. It was a happy year for us, most of the time and we hope you have enjoyed reading our humble newsletter. If you continue to give us your support we should be able to reach another 20 years without fail. I don't know about myself but TWAUG should.

ATARI CLASSICS

The Magazine for the Dedicated 8-Bit User

ENERGIZE YOUR 8-BIT: Plug into AC! ATARI CLASSICS, that is!
SUBSCRIBE NOW !!
Rates: Foreign (All) 3rd Class Mail $32 for 1 year
Europe/Mediterranean AIRMAIL $38 for 1 year
Payment by I.O.M or VISA - MASTERCARD
(Credit Card orders $2 processing fee under $100)

ATARI CLASSICS, 179 Sproul Road/RT.352, Frazer, PA 19355-1958
or Phone 313-973-8825.

THE OL'HACKERS ATARI USER GROUP INC.

O.H.A.U.G. is an all 8-bit user group in the State of New York, they are producing a bi-monthly first class informative newsletter on disk.

The disk is double sided full of news, views, articles and bonus games and/or utilities. The disk has its own printing utility which you can use to read the content of the disk on screen or make hard copies.

A large PD Library is also available.

Some of the T.W.A.U.G. members are contributing to the OL'Hackers newsletter and the OL'Hackers are contributing to the T.W.A.U.G. missive.

For more information on how to contribute to the newsletter write to:

O.H.A.U.G.
c/o A. Pignato
3375 Ocean Harbor Drive
Oceanside, N.Y.11572
U.S.A.

ULTIFONT

ULTIFONT has been written exclusively for T.W.A.U.G... and it is only available from our library, for £4.95.

It is one the most powerful character set editor written for the Atari. It fully supports the Atari 4-colour text modes and allows you to edit two adjacent characters as one. In addition, you may edit 2 different fonts at once and easily work between them.

It is totally joystick-driven, so you can sit back and relax, rather than hunching over the keyboard. ULTIFONT requires a colour display and 48K of memory.

Printed instructions come with the single sided disk.

More Support From RICHARD GORE

Classic US software re-releases:-

JAWBREAKER Guide your teeth around a maze chomping candy. Includes two game mazes, originally sold as two games. Double the value!

MOUSEKATTACK Plumb the levels of Rat Alley. Classic arcade action for one or two players on the screen at once.

PRICES: £4.50 each, available on disk, standard tape or Rambit Turbo Tape. Please state which.

INTRODUCTORY SPECIAL: Buy both Jawbreaker and Mousekattack for only £6.95

Software prices include p&p within the UK, overseas add £1 per order.

Still available: The YORKY 256k plug-in memory upgrade, only £50 + £2 p&p.

You may order from:

RICHARD GORE, 79 SPROTBROUGH ROAD, SPROTBROUGH, DONCASTER, DN5 8BW, ENGLAND

PHOENIX.

The new disk based news letter from Ireland, produced by Robert Paden. Robert recently had to give up producing the I.A.U newsletter after fur issues because of lack of support.

PHOENIX will be a double sided disk, side 'A' will be packed full of text files containing Articles, reviews and much much more. Side 'B' will contain a good selection of PD software.
The first 6 issues will be available from either the TWAUG, or from New Atari User PD libraries at £2.50 per issue.

If PHOENIX proves to be a success (which we hope it will), then from issue 7, it will only be available from Robert Paden himself.

Issue one of PHOENIX will hopefully be available by the end of August, or sometime in September. Why not give it a try.

MICRO DISCOUNT

Now the largest Mail Order Stockist of Atari 8 Bit Hardware & Software in the U.K.

Software available

The last Guardian
Tagalon Disks £5.95 H & S Speed Tapes £4.95
Adax Disk £5.95
Eureka Disk £5.95
Guitar Wizard Disk £10.95
Hans Kloss Disk £5.95
and many more

Hardware & Upgrade Kits

Atari Disk drives (re-cond) £100.00
Data Recorders XC12 (re-cond) Standard £28.00 Turbo £43.00
Upgrades from £18.95 to £51.95
and much, much more

To receive a 32 page catalogue you must first register your name and full postal address on our Data Base. The cost of this service is just £3.60 for 6 copies (1 every 8 weeks): £6.00 Europe: £10.00 Elsewhere. Prices on all items unless stated do not include post and packing, therefore please contact before ordering with details of your requirements to confirm availability and postage. Allow 28 days for delivery.

MICRO DISCOUNT
265 Chester Road
STREETLY
West Midlands B74 3EA
Tel:021-353-5730 or Fax:021-352-1669

NEW ATARI USER PAGE 6

The only magazine left in this country that supports the 8-bit

There is a large Public Domain Library available. Your support is needed to keep the magazine going.

It is now only available by subscription, for more details write to:

PAGE 6
P.O.BOX 54
STAFFORD ST16 1TB